IDENTIFICATION

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

TABLE OF CONTENTS

```
                                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.
```

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

5. OPERATION PROCEDURE

5.1 OPERATIONAL SWITCHES

NO SWITCHES ARE USED

5.2 FUNCTION ABSTRACTS

5.2.1 BEGIN SA 200

5.2.2 HLT

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

5.2.3 SCOPE

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

5.2.4 LOOP

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

5.2.5 TRAPCATCHER

THIS IS A SERIES OF INSTRUCTIONS DESIGNED TO DETECT AND ISOLATE UNEXPECTED TRAPS AND INTERRUPTS, THAT OCCUR IN THE TRAP AND INTERRUPT VECTOR AREA OF MEMORY.

THE PRINCIPAL OF THIS ROUTINE IS: THE VECTOR ENTRANCE ADDRESS POINTS TO THE NEXT SEQUENTIAL WORD WHICH WILL CONTAIN A HALT (000000) (THIS LOCATION IS ALSO THE STATUS WORD FOR THAT VECTOR ENTRANCE. BUT THIS WILL HAVE NO EFFECT ON IT, ALSO BEING THE NEXT INSTRUCTION).

IF A HALT OCCURS IN THE TRAP OR INTERRUPT VECTOR AREA, REGISTER SIX SHOULD BE EXAMINED TO DETERMINE ITS CONTENTS. THEN USE REGISTER SIX CONTENTS AS AN ADDRESS TO DETERMINE THE LOCATION THE PROGRAM WAS AT, WHEN THE INTERRUPT OR

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

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


                                6.      ERRORS

                                6.1     ALL ERRORS WILL CAUSE A HALT.

                                6.2     ERROR RECOVERY

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

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

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

                                7.      RESTRICTIONS

                                7.1     STARTING RESTRICTION

                                        THE PROGRAM MUST BE STARTED AT 200.

                                7.2     OPERATIONAL RESTRICTION

                                        NONE


                                8.      PROGRAM DESCRIPTION

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

```
                                        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                                 $ENO=1
 000001                                 $SN=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                                 ITRAP5=4
 000004                                 ERRVEC=4                ;RESERVED INST AND ILLEGAL ADDRESSES
 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
 000007                                 RESINST=000007          ;A RESERVED INSTRUCTION
 000010                                 RESVEC=10               ;RESERVED INSTRUCTION TRAP VECTOR
 004700                                 ILLA=004700
 000100                                 ILLB=100
```

```
                177776                          CC=177776
                000006                          R6=%6
                000005                          P%=%5

                076000                          MED=076000              ;MAINT. EXAM & DEPOSIT

                000200
000200  000137  001102                          .=200
                000300                          JMP     BEGIN
                                                .=300
                                .SBTTL  ACT11 HOOKS

                                                ;*******************************************************************
                                ;HOOKS REQUIRED BY ACT11
                000300                          $SVPC=.                 ;SAVE PC
                000046                          .=46
000046  021520                                  $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF SENDAD IN .$EOP
                000052                          .=52
000052  000000                                  .WORD   0               ;;2)SET LOC.52 TO ZERO
                000300                          .=$SVPC                 ;; RESTORE PC
                                .SBTTL  APT MAILBOX-ETABLE

                                                ;*******************************************************************
                                .EVEN
000300                          $MAIL:                  ;;APT MAILBOX
000300  000000  000000          $MSGTY: .WORD   AMSGTY  ;;MESSAGE TYPE CODE
000302  000000  000000          $FATAL: .WORD   AFATAL  ;;FATAL ERROR NUMBER
000304  000000  000000          $TESTN: .WORD   ATESTN  ;;TEST NUMBER
000306  000000  000000          $PASS:  .WORD   APASS   ;;PASS COUNT
000310  000000  000000          $DEVCT: .WORD   ADEVCT  ;;DEVICE COUNT
000312  000000  000000          $UNIT:  .WORD   AUNIT   ;;I/O UNIT NUMBER
000314  000000  000000          $MSGAD: .WORD   AMSGAD  ;;MESSAGE ADDRESS
000316  000000  000000          $MSGLG: .WORD   AMSGLG  ;;MESSAGE LENGTH
000320                          $ETABLE:                ;;APT ENVIRONMENT TABLE
000320  000     000             $ENV:   .BYTE   AENV    ;;ENVIRONMENT BYTE
000321  000     000             $ENVM:  .BYTE   AENVM   ;;ENVIRONMENT MODE BITS
000322  000000  000000          $SWREG: .WORD   ASWREG  ;;APT SWITCH REGISTER
000324  000000  000000          $USWR:  .WORD   AUSWR   ;;USER SWITCHES
000326  000000  000000          $CPUOP: .WORD   ACPUOP  ;;CPU TYPE,OPTIONS
                                ;*                      BITS 15-11=CPU TYPE
                                ;*                              11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
                                ;*                              11/70=06,PDQ=07,Q=10
                                ;*                      BIT 10=REAL TIME CLOCK
                                ;*                      BIT  9=FLOATING POINT PROCESSOR
                                ;*                      BIT  8=MEMORY MANAGEMENT
000330                          $ETEND:
                                .MEXIT
                                .SBTTL  APT PARAMETER BLOCK

                                                ;*******************************************************************
                                ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
                                                ;*******************************************************************
                000330                          .$X=.   ;;SAVE CURRENT LOCATION
                000024                          .=24    ;;SET POWER FAIL TO POINT TO START OF PROGRAM
000024  000200                                  200     ;;FOR APT START UP
```



```
                                .SBTTL  APT PARAMETER BLOCK

                000044                          .=44    ;;POINT TO APT INDIRECT ADDRESS PNTR.
000044  000330                                  $APTHDR ;;POINT TO APT HEADER BLOCK
                000330                          .=.$X   ;;RESET LOCATION COUNTER
                                                ;*******************************************************************
                                ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                                ;INTERFACE SPEC.

000330                          $APTHD:
000330  000000  000000          $HIBTS: .WORD   0       ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
000332  000300  000300          $MBADR: .WORD   $MAIL   ;;ADDRESS OF APT MAILBOX (BITS 0-15)
000334  000002  000002          $TSTM:  .WORD   2       ;;RUN TIM OF LONGEST TEST
000336  000002  000002          $PASTM: .WORD   2       ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
000340  000000  000000          $UNITM: .WORD   0       ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
000342  000014  000014                  .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)

                000334                  $TSTNM=$TESTN
                000302                  $ERROR=$FATAL
```

```
  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-DOKDB-A  PDP 11/6X TRAPS TEST.
  387  001034  042504  026503  030461
  388  001042  042055  045521  041104
  389  001050  040455  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,@#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     $TSTNM
  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    @#TPS           ;TTY READY?
  419  001206  100375                          BPL     BGN2
  420  001210  112037  177566                  MOVB    (R0)+,@#TPB             ;PRINT CHARACTER
  421  001214  001372                          BNE     BGN2                    ;PRINT NEXT ONE IF NOT DONE
  422
  423  001216  105737  177564          BGN2A:  TSTB    @#TPS           ;WAIT FOR DONE
  424  001222  100375                          BPL     BGN2A
```



```
  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     @#$TESTN
  432  001232  012706  021774                  MOV     #BUFF,%6                ;STACK POINTER SETUP
  433  001236  012737  001264  000010          MOV     #RETA,RESVEC    ;LOAD RESERVED INST. TRAP VECTOR
  434  001244  005037  000012                  CLR     RESVEC+2        ;AND STATUS
  435  001250  000007                          RESINST                 ;RESERVED INSTRUCTION (SHOULT TRAP)
  436
  437  001252  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
  438  001254  012713  000001                  MOV     #1,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
  439  001260  000000                          HLT             ;ERROR: RESERVED INSTRUCTION FAILED TO TRAP
  440  001262  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
  441                                                                   ;"JMP @R0" (000110) TO GET A
  442                                                                   ;SCOPE LOOP AND HIT CONTINUE
  443  001264  010700                  RETA:   SCOPE
  444
  445
  446                                          ;*****************************************************************
  447                                          ;*TEST 2 TEST DECREMENT OF STACK POINTER ON A RESERVED INSTRUCTION TRAP
  448  001266  005237  000304          TST2:   INC     @#$TESTN                ;*****************************************************************
  449  001272  012706  021774                  MOV     #BUFF,%6                ;STACK POINTER SETUP
  450  001276  012737  001306  000010          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 TYPF ($MSGTY) TO FATAL ERROR
  456  001316  012713  000002                  MOV     #2,(R3) ;MOVE TO MAIL BOX ($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 @R0" (000110) TO GET A
  462                                                                   ;SCOPE LOOP AND HIT CONTINUE
  463  001326  010700                  1$:     SCOPE
  464
  465
  466                                          ;*****************************************************************
  467                                          ;*TEST 3 TEST THAT PROPER P.C. IS SAVED ON A RESERVED INSTRUCTION TRAP
  468  001330  005237  000304          TST3:   INC     @#$TESTN                ;*****************************************************************
  469  001334  012706  021774                  MOV     #BUFF,%6                ;STACK POINTER SETUP
  470  001340  012737  001350  000010          MOV     #RETC,RESVEC
  471  001346  000007                  INSTC:  RESINST                 ;TRAP ON THIS INSTRUCTION
  472  001350  022737  001350  021770  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 @R0" (000110) TO GET A
```

```

001372  010700                  1$:     SCOPE                           ;SCOPE LOOP AND HIT CONTINUE

                                        ;*************************************************************
                                        ;*TEST 4 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON A RESERVED INSTR. TRAP
                                        ;*************************************************************
001374  005237  000304          TST4:   INC     @#$TESTN
001400  012706  021774                  MOV     #BUFF,%6                        ;SET UP
001404  012737  001420  000010          MOV     #RETD,RESVEC
001412  005037  177776                  CLR     CC                              ;CLEAR CC AND PRIORITY
001416  000007                          RESINST                         ;TRAP ON RESERVED INSTRUCTION
001420  013700  021772          RETD:   MOV     BUFF-2,%0               ;GET SAVED STATUS & TEST FOR ALL 0'S
001424  001405                          BEQ     1$                      ;BRANCH IF 0

001426  005212                          INC     (R2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
001430  012713  000004                  MOV     #4,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
001434  000000                          HLT                             ;WRONG "OLD" PSW SAVED ON STACK
                                                                        ;WHEN A RESERVED INSTR. TRAPPED
                                                                        ;EXPECT OLD PSW=0
001436  000240                          LOOP                            ;REPLACE THIS INSTRUCTION BY A
                                                                        ;"JMP @R0" (000110) TO GET A
001440  010700                  1$:     SCOPE                           ;SCOPE LOOP AND HIT CONTINUE

001442  012706  021774                  MOV     #BUFF,%6
001446  012737  001464  000010          MOV     #RETE,RESVEC                    ;INITIALIZE THE STACK POINTER
001454  012737  000357  177776          MOV     #357,CC                 ;PRE SET THE STATUS WORD
001462  000007                          RESINST                         ;RESERVED INSTRUCTION TRAP
001464  013700  021772          RETE:   MOV     BUFF-2,%0               ;GET SAVED STATUS
001470  022700  000357                  CMP     #357,%0                 ;SAVED STATUS CORRECT?
001474  001405                          BEQ     1$                      ;BRANCH IF CORRECT

001476  005212                          INC     (R2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
001500  012713  000005                  MOV     #5,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
001504  000000                          HLT                             ;WRONG "OLD" PSW SAVED ON STACK
                                                                        ;WHEN A RESERVED INSTR. TRAPPED
001506  000240                          LOOP                            ;EXPECT OLD PSW=357
                                                                        ;REPLACE THIS INSTRUCTION BY A
                                                                        ;"JMP @R0" (000110) TO GET A
001510  010700                  1$:     SCOPE                           ;SCOPE LOOP AND HIT CONTINUE


                                        ;*************************************************************
                                        ;*TEST 5 TEST THAT "NEW" STATUS IS CORRECT ON A RESERVED INSTRUCTION TRAP
                                        ;*************************************************************
001512  005237  000304          TST5:   INC     @#$TESTN
001516  012706  021774                  MOV     #BUFF,%6
001522  012737  001544  000010          MOV     #RETF,RESVEC            ;INITIALIZE THE STACK POINTER
001530  005037  000012                  CLR     RESVEC+2                ;SET UP VECTOR
001534  012737  000357  177776          MOV     #357,CC                 ;CLEAR 'NEW' STATUS
001542  000007                          RESINST                         ;PRE SET THE STATUS WORD
001544  013700  177776          RETF:   MOV     CC,%0                   ;DO A RESERVED INTRUCTION
001550  001405                          BEQ     1$                      ;GET & TEST THE 'NEW' STATUS WORD
                                                                        ;BRANCH IF ALL 0'S
001552  005212                          INC     (R2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
001554  012713  000006                  MOV     #6,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
001560  000000                          HLT                             ;"NEW" PSW WAS INCORRECT WHEN A
```



```

                                                                        ;RESERVED INSTR. TRAPPED
001562  000240                          LOOP                            ;EXPECT NEW PSW=0
                                                                        ;REPLACE THIS INSTRUCTION BY A
                                                                        ;"JMP @R0" (000110) TO GET A
001564  005037  177776          1$:     CLR     CC                      ;SCOPE LOOP AND HIT CONTINUE
001570  010700                          SCOPE

001572  012706  021774                  MOV     #BUFF,%6                ;SET UP THE STACK POINTER
001576  012737  001620  000010          MOV     #RETG,RESVEC
001604  012737  000357  000012          MOV     #357,RESVEC+2           ;LOAD THE NEW STATUS WORD
001612  005037  177776                  CLR     CC                      ;PRE SET THE STATUS WORD
001616  000007                          RESINST                         ;DO A RESERVED INSTRUCTION
001620  013700  177776          RETG:   MOV     CC,%0                   ;GET THE 'NEW' STATUS WORD
001624  022700  000357                  CMP     #357,%0                 ;WAS 'NEW' STATUS CORRECTLY LOADED
001630  001405                          BEQ     1$                      ;BRANCH IF CORRECT

001632  005212                          INC     (R2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
001634  012713  000007                  MOV     #7,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
001640  000000                          HLT                             ;"NEW" PSW WAS INCORRECT WHEN A
                                                                        ;RESERVED INSTRUCTION TRAPPED
001642  000240                          LOOP                            ;EXPECT NEW PSW=357
                                                                        ;REPLACE THIS INSTRUCTION BY A
                                                                        ;"JMP @R0" (000110) TO GET A
001644                                                                  ;SCOPE LOOP AND HIT CONTINUE
001644  012737  000012  000010  1$:     MOV     #RESVEC+2,RESVEC                ;RESTORE RESERVED INSTRUCTION
001652  005037  000012                  CLR     RESVEC+2                ;TO HALT AT RESVEC+2
001656  010700                          SCOPE


                                        ;*************************************************************
                                        ;*TEST 6 TEST THAT A TRAP OCCURES FOR A "TRAP" INSTRUCTION
                                        ;*************************************************************
001660  005237  000304          TST6:   INC     @#$TESTN
001664  012706  021774                  MOV     #BUFF,%6                        ;STACK POINTER SETUP
001670  012737  001716  000034          MOV     #RETA1,TRAPVEC  ;LOAD TRAP VECTOR
001676  005037  000036                  CLR     TRAPVEC+2
001702  104400                          TRAP                            ;DO A TRAP INSTRUCTION

001704  005212                          INC     (R2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
001706  012713  000010                  MOV     #10,(R3)                ;MOVE TO MAIL BOX ($FATAL) THE ERROR
001712  000000                          HLT                             ;"TRAP" INSTRUCTION DID NOT TRAP
001714  000240                          LOOP                            ;REPLACE THIS INSTRUCTION BY A
                                                                        ;"JMP @R0" (000110) TO GET A
001716  010700                  RETA1:  SCOPE                           ;SCOPE LOOP AND HIT CONTINUE


                                        ;*************************************************************
                                        ;*TEST 7 TEST DECREMENT OF STACK POINTER ON A "TRAP" INSTRUCTION
                                        ;*************************************************************
001720  005237  000304          TST7:   INC     @#$TESTN
001724  012706  021774                  MOV     #BUFF,%6
001730  012737  001740  000034          MOV     #RETB1,TRAPVEC                  ;STACK POINTER SETUP
001736  104400                          TRAP                                    ;RETURN POINTER
001740  020627  021770          RETB1:  CMP     %6,#BUFF-4              ;DO A TRAP INSTRUCTION
001744  001405                          BEQ     1$                              ;TEST DECREMENT OF %6
```

```

001746 005212                 INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
001750 012713 000011          MOV     #11,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
001754 000000                 HLT                             ;STACK POINTER (R6) DID NOT DECREMENT
                                                      ;TWO WORDS WHEN "TRAP" TRAPPED
001756 000240                 LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                      ;"JMP @R0" (000110) TO GET A
                                                      ;SCOPE LOOP AND HIT CONTINUE
001760 010700          1$:    SCOPE

                                                      ;*************************************************************
                                                      ;*TEST 10 TEST THAT PROPER P.C. IS SAVED ON A "TRAP" INSTRUCTION
                                                      ;*************************************************************
001762 005237 000304   TST10: INC     @#$TESTN
001766 012706 021774          MOV     #BUFF,%6                ;STACK POINTER SETUP
001772 012737 002002 000034   MOV     #RETC1,TRAPVEC          ;RETURN FROM TRAP POINTER
002000 104400                 TRAP                            ;TRAP ON THIS INSTRUCTION
002002 022737 002002 021770 RETC1: CMP #.,BUFF-4              ;CHECK INCREMENTED P.C.
002010 001405                 BEQ     1$

002012 005212                 INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002014 012713 000012          MOV     #12,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002020 000000                 HLT                             ;WRONG "OLD" PC SAVED ON THE STACK
                                                              ;WHEN "TRAP" TRAPPED
002022 000240                 LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                      ;"JMP @R0" (000110) TO GET A
                                                      ;SCOPE LOOP AND HIT CONTINUE
002024 010700          1$:    SCOPE
                                                      ;*************************************************************
                                                      ;*TEST 11 TEST THAT "OLD" CC & PRIORITY ARE PLACED ON STACK, ON A "TRAP" TRAP
                                                      ;*************************************************************
002026 005237 000304   TST11: INC     @#$TESTN
002032 012706 021774          MOV     #BUFF,%6                ;SET UP
002036 012737 002052 000034   MOV     #RETD1,TRAPVEC          ;SET UP
002044 005037 177776          CLR     CC                      ;CLEAR CC AND PRIORITY
002050 104400                 TRAP                    ;DO A TRAP INSTRUCTION
002052 013700 021772   RETD1: MOV     BUFF-2,%0       ;GET & TEST THE SAVED STATUS
002056 001405                 BEQ     1$              ;BRANCH IF ALL 0'S

002060 005212                 INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002062 012713 000013          MOV     #13,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002066 000000                 HLT                     ;WRONG "OLD" PSW SAVED ON THE STACK
                                                      ;WHEN "TRAP" TRAPPED, EXPECT "0"
002070 000240                 LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                      ;"JMP @R0" (000110) TO GET A
                                                      ;SCOPE LOOP AND HIT CONTINUE
002072 010700          1$:    SCOPE

002074 012706 021774          MOV     #BUFF,%6        ;INITIALIZE THE STACK POINTER
002100 012737 002116 000034   MOV     #RETE1,TRAPVEC          ;SET UP
002106 012737 000357 177776   MOV     #357,CC         ;PRE SET THE STATUS WORD
002114 104400                 TRAP                    ;DO A TRAP INSTRUCTION
002116 013700 021772   RETE1: MOV     BUFF-2,%0       ;GET THE SAVED STATUS WORD
002122 022700 000357          CMP     #357,%0         ;WAS CORRECT STATUS SAVED ON THE STACK
002126 001405                 BEQ     1$              ;BRANCH IF CORRECT
```



```
002130 005212                 INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002132 012713 000014          MOV     #14,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002136 000000                 HLT                     ;WRONG "OLD" PSW SAVED ON THE STACK
                                                      ;WHEN "TRAP" TRAPPED, EXPECT "357"
002140 000240                 LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                      ;"JMP @R0" (000110) TO GET A
                                                      ;SCOPE LOOP AND HIT CONTINUE
002142 010700          1$:    SCOPE


                                                      ;*************************************************************
                                                      ;*TEST 12 TEST THAT "NEW" STATUS IS CORRECT ON A "TRAP" TRAP
                                                      ;*************************************************************
002144 005237 000304   TST12: INC     @#$TESTN
002150 012706 021774          MOV     #BUFF,%6
002154 012737 002176 000034   MOV     #RETF1,TRAPVEC
002162 005037 000036          CLR     TRAPVEC+2               ;CLEAR FUTURE PRIORITY AND CC
002166 012737 000357 177776   MOV     #357,CC         ;PRE SET THE STATUS WORD
002174 104400                 TRAP
002176 013700 177776   RETF1: MOV     CC,%0           ;GET THE 'NEW' STATUS WORD
002202 001405                 BEQ     1$              ;BRANCH IF ALL 0'S

002204 005212                 INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002206 012713 000015          MOV     #15,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002212 000000                 HLT                     ;ERROR! INCORRECT 'NEW' PSW LOADED
002214 000240                 LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                      ;"JMP @R0" (000110) TO GET A
                                                      ;SCOPE LOOP AND HIT CONTINUE
002216 005037 177776   1$:    CLR     CC              ;WHEN "TRAP" TRAPPED, EXPECT 0
002222 010700                 SCOPE

002224 012706 021774          MOV     #BUFF,%6
002230 012737 002252 000034   MOV     #RETG1,TRAPVEC
002236 012737 000357 000036   MOV     #357,TRAPVEC+2  ;LOAD 'NEW' STATUS WORD
002244 005037 177776          CLR     CC              ;PRE SET THE STATUS WORD
002250 104400                 TRAP                    ;DO A TRAP INSTRUCTION
002252 013700 177776   RETG1: MOV     CC,%0           ;GET THE 'NEW' STATUS
002256 022700 000357          CMP     #357,%0         ;IS IT CORRECT?
002262 001405                 BEQ     1$              ;BRANCH IF CORRECT

002264 005212                 INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002266 012713 000016          MOV     #16,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002272 000000                 HLT                     ;INCORRECT "NEW" PSW LOADED WHEN
                                                      ;"TRAP" TRAPPED, EXPECT 357
002274 000240                 LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                      ;"JMP @R0" (000110) TO GET A
                                                      ;SCOPE LOOP AND HIT CONTINUE
002276 005037 177776   1$:    CLR     CC
002302 010700                 SCOPE

                                                      ;*************************************************************
                                                      ;*TEST 13 TEST THAT ALL COMBINATION OF "TRAP" WILL CAUSE A TRAP
                                                      ;*************************************************************
002304 005237 000304   TST13: INC     @#$TESTN
002310 012737 104400 002330   MOV     #TRAP,RB1               ;INITIALIZE BASE TRAP INSTRUCTION
002316 012737 002344 000034   MOV     #RA1,34                 ;RETURN FROM TRAP TO RA1
002324 012706 021774   RC1:   MOV     #BUFF,%6                ;SET UP STACK POINTER
```

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

```
002330  104400          RB1:    TRAP                            ;TRAP INST WILL BE MODIFIED TO TRAP+377
                                ;
002332  005212                  INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002334  012713  000017          MOV     #17,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002340  000000                  HLT                             ;"TRAP" INSTRUCTION HELD IN "RB1"
                                                                ;FAILED TO TRAP
002342  000240                  LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
002344  005737  002330  RA1:    INC     RB1                     ;INCREMENT TRAP INSTRUCTION
002350  022737  104777  002330  CMP     #104777,RB1             ;TRAP+377 TO UPPER LIMIT
002356  103362                  BHIS    RC1                     ;HAVE WE TESTED ALL
                                                                ;YES
002360  012737  000036  000034  MOV     #36,TRAPVEC     ;RESTORE TRAP VECTOR TO
002366  005037  000036          CLR     TRAPVEC+2       ;HALT AT 36
002372  010700                  SCOPE
```


T14 TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION

```
                                ;*****************************************************************
                                ;*TEST 14 TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION
                                ;*****************************************************************
002374  005237  000304  TST14:  INC     @#$TESTN
002400  012706  021774          MOV     #BUFF,%6                ;STACK POINTER SETUP
002404  012737  002432  000020  MOV     #RETA2,IOTVEC           ;RETURN LOCATION
002412  005037  000022          CLR     IOTVEC+2
002416  000004                  IOT
002420  005212                  INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002422  012713  000020          MOV     #20,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002426  000000                  HLT                             ;"IOT" INSTRUCTION FAILED TO TRAP
002430  000240                  LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
002432  010700          RETA2:  SCOPE                   ;SCOPE LOOP AND HIT CONTINUE

                                ;*****************************************************************
                                ;*TEST 15 TEST DECREMENT OF STACK POINTER ON AN "IOT" TRAP
                                ;*****************************************************************
002434  005237  000304  TST15:  INC     @#$TESTN
002440  012706  021774          MOV     #BUFF,%6                ;STACK POINTER SETUP
002444  012737  002454  000020  MOV     #RETB2,IOTVEC           ;RETURN POINTER
002452  000004                  IOT
002454  020627  021770  RETB2:  CMP     %6,#BUFF-4              ;TEST DECREMENT OF %6
002460  001405                  BEQ     1$
002462  005212                  INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002464  012713  000021          MOV     #21,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002470  000000                  HLT                             ;STACK POINTER DID NOT DECREMENT BY
                                                                ;TWO WORDS ON AN "IOT" TRAP
002472  000240                  LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
002474  010700          1$:     SCOPE                   ;SCOPE LOOP AND HIT CONTINUE

                                ;*****************************************************************
                                ;*TEST 16 TEST THAT PROPER P.C. IS SAVED ON AN "IOT" TRAP
                                ;*****************************************************************
002476  005237  000304  TST16:  INC     @#$TESTN
002502  012706  021774          MOV     #BUFF,%6                ;STACK POINTER SETUP
002506  012737  002516  000020  MOV     #RETC2,IOTVEC           ;RETURN FROM TRAP POINTER
002514  000004                  IOT                             ;TRAP ON THIS INSTRUCTION
002516  022737  002516  021770  RETC2:  CMP     .,BUFF-4                ;CHECK FOR INCREMENTED P.C.
002524  001405                  BEQ     1$
002526  005212                  INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002530  012713  000022          MOV     #22,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002534  000000                  HLT                             ;WRONG "OLD" PC WAS SAVED ON
                                                                ;STACK WHEN "IOT" TRAPPED
002536  000240                  LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
002540  010700          1$:     SCOPE                   ;SCOPE LOOP AND HIT CONTINUE

                                ;*****************************************************************
```

```
                                        ;*TEST 17 TEST THAT "OLD" CC AND PRIORITY ARE PLACED ON STACK, ON AN "IOT" TRAP
                                        ;*************************************************************
002542  005237  000304          TST17:  INC     @#$TESTN
002546  012706  021774                  MOV     #BUFF,%6                ;SET UP
002552  012737  002566  000020          MOV     #RETD2,IOTVEC           ;SET UP
002560  005037  177776                  CLR     CC                      ;CLEAR CC AND PRIORITY
002564  000004                          IOT                             ;TRAP
002566  013700  021772          RETD2:  MOV     BUFF-2,%0       ;GET & TEST SAVED STATUS
002572  001405                          BEQ     1$
                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002576  012713  000023                  MOV     #23,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002602  000000                          HLT                     ;WRONG "OLD" PSW SAVED ON STACK
                                        ;WHEN "IOT" TRAPPED, EXPECT 0
002604  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
                                        ;SCOPE LOOP AND HIT CONTINUE
002606  010700                  1$:     SCOPE
002610  012706  021774                  MOV     #BUFF,%6                ;SET UP
002614  012737  002632  000020          MOV     #RETE2,IOTVEC           ;SET UP
002622  012737  000357  177776          MOV     #357,CC         ;PRE SET STATUS
002630  000004                          IOT
002632  013700  021772          RETE2:  MOV     BUFF-2,%0       ;GET SAVED STATUS
002636  022700  000357                  CMP     #357,%0         ;SAVED STATUS CORRECT?
002642  001404                          BEQ     1$              ;BRANCH IF CORRECT
002644  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002646  012713  000024                  MOV     #24,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002652  000000                          HLT                     ;WRONG "OLD" PSW SAVED ON STACK
                                        ;WHEN "IOT" TRAPPED, EXPECT 357
002654  000240                  1$:     LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
                                        ;SCOPE LOOP AND HIT CONTINUE
002656  010700                          SCOPE

                                        ;*************************************************************
                                        ;*TEST 20 TEST THAT "NEW" STATUS IS CORRECT ON AN "IOT" TRAP
                                        ;*************************************************************
002660  005237  000304          TST20:  INC     @#$TESTN
002664  012706  021774                  MOV     #BUFF,%6
002670  012737  002712  000020          MOV     #RETF2,IOTVEC
002676  005037  000022                  CLR     IOTVEC+2                ;CLEAR FUTURE PRIORITY AND CC
002702  012737  030357  177776          MOV     #30357,CC       ;PRE SET STATUS
002710  000004                          IOT
002712  013700  177776          RETF2:  MOV     CC,%0           ;GET & TEST 'NEW' STATUS
002716  001405                          BEQ     1$              ;BRANCH IF CORRECT
002720  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
002722  012713  000025                  MOV     #25,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
002726  000000                          HLT                     ;WRONG "NEW" PSW LOADED WHEN
                                        ;"IOT" TRAPPED, EXPECT 0
002730  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
                                        ;SCOPE LOOP AND HIT CONTINUE
002732  005037  177776          1$:     CLR     CC
```



```
002736  010700                          SCOPE

002740  012706  021774                  MOV     #BUFF,%6
002744  012737  002766  000020          MOV     #RETG2,IOTVEC
002752  012737  000357  000022          MOV     #357,IOTVEC+2   ;LOAD 'NEW' STATUS
002760  005037  177776                  CLR     CC              ;PRE SET STATUS
002764  000004                          IOT
002766  013700  177776          RETG2:  MOV     CC,%0           ;GET THE 'NEW' STATUS
002772  022700  000357                  CMP     #357,%0         ;IS 'NEW' STATUS CORRECT
002776  001405                          BEQ     1$              ;BRANCH IF CORRECT

003000  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
003002  012713  000026                  MOV     #26,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
003006  000000                          HLT                     ;WRONG "NEW" PSW LOADED WHEN
                                        ;"IOT" TRAPPED, EXPECT 357
003010  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
                                        ;SCOPE LOOP AND HIT CONTINUE
003012                          1$:

003012  012737  000022  000020          MOV     #22,IOTVEC      ;RESTORE IOT TRAP VECTOR
003020  005037  000022                  CLR     IOTVEC+2        ;TO HALT AT 22
003024  010700                          SCOPE
```

```
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     @#$TESTN
1071  003704  012706  021774                  MOV     #BUFF,%6                ;SET UP
1072  003710  012737  003724  000014          MOV     #RETD4,BPTVEC   ;LOAD VECTOR
1073  003716  005037  177776                  CLR     CC                      ;CLEAR CC AND PRIORITY
1074  003722  000003                          BPT
1075  003724  013700  021772          RETD4:  MOV     BUFF-2,%0       ;GET SAVED STATUS OFF STACK
1076  003730  001405                          BEQ     1$              ;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 @R0" (000110) TO GET A
1084                                                                  ;SCOPE LOOP AND HIT CONTINUE
1085  003744  010703                  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 @R0" (000110) TO GET A
1101                                                                  ;SCOPE LOOP AND HIT CONTINUE
1102  004014  010700                  1$:     SCOPE
1103
1104                                            ;*************************************************************
1105                                            ;*TEST 33 TEST THAT "NEW" STATUS IS CORRECT ON A "BPT" TRAP
1106                                            ;*************************************************************
1107  004016  005237  000304          TST33:  INC     @#$TESTN
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 @R0" (000110) TO GET A
1122                                                                  ;SCOPE LOOP AND HIT CONTINUE
```



```
1123  004070  005037  177776          1$:     CLR     CC
1124  004074  010700                          SCOPE
1125
1126  004076  012706  021774                  MOV     #BUFF,%6
1127  004102  012737  004124  000014          MOV     #RETG4,BPTVEC
1128  004110  012737  000357  000016          MOV     #357,BPTVEC+2   ;LOAD "NEW" STATUS
1129  004116  005037  177776                  CLR     CC              ;PRE SET STATUS
1130  004122  000003                          BPT
1131  004124  013700  177776          RETG4:  MOV     CC,%0           ;GET THE "NEW" STATUS
1132  004130  022700  000357                  CMP     #357,%0         ;IS IT CORRECT
1133  004134  001405                          BEQ     1$              ;BRANCH IF CORRECT
1134
1135  004136  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1136  004140  012713  000045                  MOV     #45,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1137  004144  000000                          HLT                     ;WRONG "NEW" PSW LOADED ON
1138                                                                  ;"BPT" TRAP, EXPECT 357
1139  004146  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
1140                                                                  ;"JMP @R0" (000110) TO GET A
1141                                                                  ;SCOPE LOOP AND HIT CONTINUE
1142  004150  005037  177776          1$:     CLR     CC
1143
1144  004154  012737  000016  000014          MOV     #16,BPTVEC      ;RESTORE "IRT" TRAP
1145  004162  005037  000016                  CLR     16              ;TO HALT AT 16
1146  004166  010700                          SCOPE
1147
```

```
                                        ;*****************************************************************
                                        ;*TEST 34 TEST THAT A TRAP OCCURS ON AN ILLEGAL INSTRUCTION - JMP R0
                                        ;*****************************************************************
004170  005237  000304          TST34:  INC     @#$TESTN
004174  012706  021774                  MOV     #BUFF,%6        ;STACK POINTER SETUP
004200  012737  004226  000004          MOV     #RETA5,ERRVEC     ;RETURN LOCATION
004206  005037  000006                  CLR     ERRVEC+2
004212  000100                          JMP     %0              ;ILLEGAL INSTRUCTION, SHOULD TRAP
                                        ;
004214  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
004216  012713  000046                  MOV     #46,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
004222  000000                          HLT                     ;ILLEGAL INSTR. DID NOT TRAP
004224  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
004226  010700                  RETA5:    SCOPE
                                        ;
                                        ;*****************************************************************
                                        ;*TEST 35 TEST DECREMENT OF STACK POINTER ON AN ILLEGAL INSTRUCTION TRAP
                                        ;*****************************************************************
004230  005237  000304          TST35:  INC     @#$TESTN
004234  012706  021774                  MOV     #BUFF,%6        ;STACK POINTER SETUP
004240  012737  004250  000004          MOV     #RETB5,ERRVEC     ;RETURN POINTER
004246  000100                          JMP     %0              ;RESERVED INSTRUCTION
004250  020627  021770          RETB5:  CMP     %6,#BUFF-4        ;TEST DECREMENT OF %6
004254  001405                          BEQ     1$
                                        ;
004256  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
004260  012713  000047                  MOV     #47,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
004264  000000                          HLT                     ;STACK POINTER DID NOT DECREMENT
                                        ;                       ;BY TWO WORDS ON ILLEGAL INSTR. TRAP
004266  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
004270  010700                  1$:       SCOPE
                                        ;
                                        ;*****************************************************************
                                        ;*TEST 36 TEST THAT PROPER P.C. IS SAVED ON AN ILLEGAL INSTRUCTION TRAP
                                        ;*****************************************************************
004272  005237  000304          TST36:  INC     @#$TESTN
004276  012706  021774                  MOV     #BUFF,%6        ;STACK POINTER SETUP
004302  012737  004312  000004          MOV     #RETC5,ERRVEC     ;RETURN FROM TRAP POINTER
004310  000100                          JMP     %0              ;TRAP ON THIS INSTRUCTION
004312  027737  004312  021770  RETC5:  CMP     #.,BUFF-4       ;CHECK FOR INCREMENTED P.C.
004320  001405                          BEQ     1$
                                        ;
004322  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
004324  012713  000050                  MOV     #50,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
004330  000000                          HLT                     ;WRONG "OLD" PC SAVED ON STACK
                                        ;                       ;ON ILLEGAL INSTR. TRAP
004332  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
004334  010700                  1$:     SCOPE
```



```
                                        ;*****************************************************************
                                        ;*TEST 37 TEST THAT "OLD" CC AND PRIORITY ARE SAVED ON AN ILLEGAL INSTR. TRAP
                                        ;*****************************************************************
004336  005237  000304          TST37:  INC     @#$TESTN
004342  012706  021774                  MOV     #BUFF,%6        ;SET UP
004346  012737  004362  000004          MOV     #RETD5,ERRVEC   ;SET UP
004354  005037  177776                  CLR     CC              ;CLEAR CC AND PRIORITY
004360  000100                          JMP     %0              ;TRAP
004362  013700  021772          RETD5:  MOV     BUFF-2,%0       ;GET THE SAVED STATUS
004366  001405                          BEQ     1$              ;BRANCH IF ALL 0'S
                                        ;
004370  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
004372  012713  000051                  MOV     #51,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
004376  000000                          HLT                     ;ERROR! SAVED PSW IS INCORRECT
                                        ;                       ;ON ILLEGAL INSTR. TRAP, EXPECT 0
004400  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
004402  010700                  1$:     SCOPE
                                        ;
004404  012706  021774                  MOV     #BUFF,%6        ;SET UP
004410  012737  004426  000004          MOV     #RETE5,ERRVEC     ;SET UP
004416  012737  000357  177776          MOV     #357,CC         ;PRE SET STATUS
004424  000101                          JMP     %1
004426  013700  021772          RETE5:  MOV     BUFF-2,%0       ;GET THE SAVED STATUS
004432  022700  000357                  CMP     #357,%0         ;IS IT CORRECT
004436  001405                          BEQ     1$              ;BRANCH IF SAVED STATUS IS CORRECT
                                        ;
004440  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
004442  012713  000052                  MOV     #52,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
004446  000000                          HLT                     ;ERROR! SAVED PSW IS INCORRECT
                                        ;                       ;ON ILLEGAL INSTR. TRAP, EXPECT 357
004450  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
004452  010700                  1$:     SCOPE
                                        ;
                                        ;*****************************************************************
                                        ;*TEST 40 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP
                                        ;*****************************************************************
004454  005237  000304          TST40:  INC     @#$TESTN
004460  012706  021774                  MOV     #BUFF,%6
004464  012737  004506  000004          MOV     #RETF5,ERRVEC
004472  005037  000006                  CLR     ERRVEC+2        ;CLEAR FUTURE PRIORITY AND CC
004476  012737  000357  177776          MOV     #357,CC         ;PRE SET STATUS
004504  000100                          JMP     %0
004506  013700  177776          RETF5:  MOV     CC,%0           ;GET & TEST "NEW" STATUS
004512  001405                          BEQ     1$
                                        ;
004514  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
004516  012713  000053                  MOV     #53,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
004522  000000                          HLT                     ;ERROR! "NEW" PSW IS INCORRECT
                                        ;                       ;ON ILLEGAL INSTR. TRAP, EXPECT 0
004524  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
```

```
 1260  004526  005237  177776          1$:     CLR     CC
 1261  004532  010700                          SCOPE
 1262
 1263  004534  012706  021774                  MOV     #BUFF,%6
 1264  004540  012737  004562  000004          MOV     #RETG5,ERRVEC
 1265  004546  012737  000357  000006          MOV     #357,ERRVEC+2   ;LOAD 'NEW' STATUS
 1266  004554  005037  177776                  CLR     CC              ;PRE SET STATUS
Z 1267  004560  000100                          JMP     %0
 1268  004562  013700  177776          RETG5:  MOV     CC,%0           ;GET THE 'NEW' STATUS
 1269  004566  022700  000357                  CMP     #357,%0         ;IS IT CORRECT
 1270  004572  001405                          BEQ     1$              ;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          1$:     CLR     CC
 1280  004612  010700                          SCOPE
```



```
 1281
 1282                                          ;*************************************************************
 1283                                          ;*TEST 41 TEST THAT A TRAP OCCURS ON ALL ILLEGAL INSTRUCTION - JSR R0,R0
 1284                                          ;*************************************************************
 1285  004614  005237  000304          TST41:  INC     @#$TESTN
 1286  004620  012706  021774                  MOV     #BUFF,%6        ;STACK POINTER SETUP
 1287  004624  012737  004652  000004          MOV     #RETH5,ERRVEC     ;RETURN LOCATION
 1288  004632  005037  000006                  CLR     ERRVEC+2
Z 1289  004636  004000                          JSR     %0,%0           ;ILLEGAL INSTRUCTION (SHOULD TRAP)
 1290
 1291  004640  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 1292  004642  012713  000055                  MOV     #55,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 1293  004646  000000                          HLT                     ;ILLEGAL INSTRUCTION FAILED TO TRAP
 1294  004650  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 1295                                                                  ;"JMP @R0" (000110) TO GET A
 1296                                                                  ;SCOPE LOOP AND HIT CONTINUE
 1297  004652  010700          RETH5:  SCOPE
 1298
 1299                                  ;TEST DECREMENT OF STACK POINTER ON A TRAP OPERATION
 1300  004654  012706  021774                  MOV     #BUFF,%6        ;STACK POINTER SETUP
 1301  004660  012737  004670  000004          MOV     #RETJ,ERRVEC     ;RETURN POINTER
Z 1302  004666  004000                          JSR     %0,%0           ;RESERVED INSTRUCTION
 1303  004670  020627  021770          RETJ:   CMP     %6,#BUFF-4      ;TEST DECREMENT OF %6
 1304  004674  001405                          BEQ     1$
 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 @R0" (000110) TO GET A
 1312                                                                  ;SCOPE LOOP AND HIT CONTINUE
 1313  004710  010700          1$:       SCOPE
 1314
 1315
 1316                                          ;*************************************************************
 1317                                          ;*TEST 42 TEST THAT PROPER P.C. IS SAVED ON AN ILLEGAL INSTR. TRAP
 1318  004712  005237  000304          TST42:  INC     @#$TESTN
 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 @R0" (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
                                              ;*************************************************************
```

```
  1337  004756  005237  000304          TST43:  INC     @#STESTN
  1338  004762  012706  021774                  MOV     #BUFF,%6        ;SET UP
  1339  004766  012737  005402  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     1$              ;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                     ;ERROR! INCORRECT PSW SAVED ON STACK
  1348                                                                  ;ON ILLEGAL INSTR. TRAP, EXPECT 0
  1349  005020  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
  1350                                                                  ;"JMP @R0" (000110) TO GET A
  1351                                                                  ;SCOPE LOOP AND HIT CONTINUE
  1352  005022  010700                  1$:     SCOPE
  1353
  1354  005024  012706  021774                  MOV     #BUFF,%6        ;SET UP
  1355  005030  012737  005046  000004          MOV     #RETM,ERRVEC     ;SET UP
  1356  005036  012737  000357  177776          MOV     #357,CC         ;PRE SET STATUS
Z 1357  005044  004000                          JSR     %0,%0
  1358  005046  013700  021772          RETM:   MOV     BUFF-2,%0       ;GET SAVED STATUS
  1359  005052  022700  000357                  CMP     #357,%0         ;IS IT CORRECT
  1360  005056  001405                          BEQ     1$
  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                     ;ERROR! INCORRECT PSW SAVED ON STACK
  1365                                                                  ;ON ILLEGAL INSTR. TRAP, EXPECT 357
  1366  005070  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
  1367                                                                  ;"JMP @R0" (000110) TO GET A
  1368                                                                  ;SCOPE LOOP AND HIT CONTINUE
  1369  005072  010700                  1$:     SCOPE
  1370
  1371                                  ;*********************************************************************
  1372                                  ;*TEST 44 TEST THAT "NEW" STATUS IS CORRECT ON AN ILLEGAL INSTR. TRAP
  1373                                  ;*********************************************************************
  1374  005074  005237  000304          TST44:  INC     @#STESTN
  1375  005100  012706  021774                  MOV     #BUFF,%6
  1376  005104  012737  005126  000004          MOV     #RETN,ERRVEC
  1377  005112  005037  000006                  CLR     ERRVEC+2         ;CLEAR FUTURE PRIORITY AND CC
  1378  005116  012737  000357  177776          MOV     #357,CC         ;PRE SET STATUS
Z 1379  005124  004000                          JSR     %0,%0
  1380  005126  013700  177776          RETN:   MOV     CC,%0           ;GET & TEST "NEW" STATUS
  1381  005132  001405                          BEQ     1$              ;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                     ;ERROR! "NEW" PSW IS INCORRECT
  1386                                                                  ;ON ILLEGAL INSTR. TRAP
  1387  005144  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
  1388                                                                  ;"JMP @R0" (000110) TO GET A
  1389                                                                  ;SCOPE LOOP AND HIT CONTINUE
  1390  005146  005037  177776          1$:     CLR     CC
  1391  005152  010700                          SCOPE
  1392
```



```
  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     1$              ;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" PSW IS INCORRECT
  1405                                                                  ;ON ILLEGAL INSTR. TRAP
  1406  005224  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
  1407                                                                  ;"JMP @R0" (000110) TO GET A
  1408                                                                  ;SCOPE LOOP AND HIT CONTINUE
  1409  005226  005037  177776          1$:     CLR     CC
  1410  005232  010700                          SCOPE
  1411
  1412
  1413                                  ;*********************************************************************
  1414                                  ;*TEST 45 TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS (ODD)
  1415                                  ;*********************************************************************
  1416  005234  005237  000304          TST45:  INC     @#STESTN
  1417  005240  012706  021774                  MOV     #BUFF,%6        ;STACK POINTER SETUP
  1418  005244  012737  005274  000004          MOV     #RETP,ERRVEC     ;RETURN LOCATION
  1419  005252  005037  000006                  CLR     ERRVEC+2
  1420  005256  005737  000001                  TST     1               ;ILL.ADRS. (ODD ADDRESS ON WORD INST.)
  1421
  1422  005262  005212                          INC     (R2)            ;SET MESSAGE TYPE ($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 @R0" (000110) TO GET A
  1427                                                                  ;SCOPE LOOP AND HIT CONTINUE
  1428  005274  010700                  RETP:   SCOPE
  1429
```

```
1430
1431
1432                                    ;*****************************************************************
1433                                    ;*TEST 46 TEST DECREMENT OF STACK POINTER ON AN ODD ADDRESS TRAP
1434                                    ;*****************************************************************
1434  005276  005237  000304   TST46:  INC     @#$TESTN
1435  005302  012706  021774           MOV     #BUFF,%6        ;STACK POINTER SETUP
1436  005306  012737  005320  000004   MOV     #RETQ,ERRVEC     ;RETURN POINTER
1437  005314  005737  000001           TST 1                   ;RESERVED INSTRUCTION
1438  005320  026627  021770   RETQ:   CMP     %6,#BUFF-4      ;TEST DECREMENT OF %6
1439  005324  001405                   BEQ     1$
1440
1441  005326  005212                   INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1442  005330  012713  000065           MOV     #65,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1443  005334  000000                   HLT                     ;R6 DID NOT DECREMENT BY 2 WORDS
1444                                                            ;ON AN OCTAL ADDRESS TRAP
1445  005336  000240                   LOOP                    ;REPLACE THIS INSTRUCTION BY A
1446                                                            ;"JMP @R0" (000110) TO GET A
1447                                                            ;SCOPE LOOP AND HIT CONTINUE
1448  005340  010700           1$:     SCOPE
1449
1450                                    ;*****************************************************************
1451                                    ;*TEST 47 TEST THAT PROPER P.C. IS SAVED ON AN ODD ADDRESS TRAP
1452                                    ;*****************************************************************
1453  005342  005237  000304   TST47:  INC     @#$TESTN
1454  005346  012706  021774           MOV     #BUFF,%6        ;STACK POINTER SETUP
1455  005352  012737  005364  000004   MOV     #RETR,ERRVEC     ;RETURN FROM TRAP POINTER
1456  005360  005737  000001           TST 1                   ;TRAP ON THIS INSTRUCTION
1457  005364  022737  005364  021770   RETR:   CMP     #.,BUFF-4       ;CHECK FOR INCREMENTED P.C.
1458  005372  001405                   BEQ     1$
1459
1460  005374  005212                   INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1461  005376  012713  000066           MOV     #66,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1462  005402  000000                   HLT                     ;WRONG "OLD" PC SAVED ON STACK
1463                                                            ;ON ODD ADDRESS TRAP
1464  005404  000240                   LOOP                    ;REPLACE THIS INSTRUCTION BY A
1465                                                            ;"JMP @R0" (000110) TO GET A
1466                                                            ;SCOPE LOOP AND HIT CONTINUE
1467  005406  010700           1$:     SCOPE
1468
1469
1470                                    ;*****************************************************************
1471                                    ;*TEST 50 TEST THAT "OLD" CC & PRIORITY ARE SAVED ON AN ODD ADDRESS TRAP
1472  005410  005237  000304   TST50:  INC     @#$TESTN        ;*****************************************************************
1473  005414  012706  021774           MOV     #BUFF,%6        ;SET UP
1474  005420  012737  005436  000004   MOV     #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     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 ($MSGTY) TO FATAL ERROR
1481  005446  012713  000067           MOV     #67,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1482  005452  000000                   HLT                     ;ERROR! SAVED PSW IS INCORRECT
1483  005454  000240                   LOOP                    ;REPLACE THIS INSTRUCTION BY A
1484                                                            ;"JMP @R0" (000110) TO GET A
1485                                                            ;SCOPE LOOP AND HIT CONTINUE
```



```
1486  005456  010700           1$:     SCOPE
1487
1488  005460  012706  021774           MOV     #BUFF,%6        ;SET UP
1489  005464  012737  005504  000004   MOV     #RETT,ERRVEC     ;SET UP
1490  005472  012737  000357  177776   MOV     #357,CC         ;PRE SET STATUS
1491  005500  005737  000001           TST     1
1492  005504  013700  021772   RETT:   MOV     BUFF-2,%0       ;GET THE SAVED STATUS OFF STACK
1493  005510  022700  000357           CMP     #357,%0         ;IS IT CORRECT
1494  005514  001405                   BEQ     1$              ;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                     ;ERROR! SAVED PSW ON STACK IS INCORRECT
1499  005526  000240                   LOOP                    ;REPLACE THIS INSTRUCTION BY A
1500                                                            ;"JMP @R0" (000110) TO GET A
1501                                                            ;SCOPE LOOP AND HIT CONTINUE
1502  005530  010700           1$:     SCOPE
1503
1504                                    ;*****************************************************************
1505                                    ;*TEST 51 TEST THAT "NEW" STATUS IS CORRECT ON AN ODD ADDRESS TRAP
1506                                    ;*****************************************************************
1507  005532  005237  000304   TST51:  INC     @#$TESTN
1508  005536  012706  021774           MOV     #BUFF,%6
1509  005542  012737  005566  000004   MOV     #RETU,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     1$              ;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                     ;ERROR! "NEW" PSW IS INCORRECT
1519  005604  000240                   LOOP                    ;REPLACE THIS INSTRUCTION BY A
1520                                                            ;"JMP @R0" (000110) TO GET A
1521                                                            ;SCOPE LOOP AND HIT CONTINUE
1522  005606  005037  177776   1$:     CLR     CC
1523  005612  010700                   SCOPE
1524
1525  005614  012706  021774           MOV     #BUFF,%6
1526  005620  012737  005640  000004   MOV     #RETV,ERRVEC
1527  005626  012737  000357  000006   MOV     #357,ERRVEC+2   ;LOAD "NEW" STATUS
1528  005634  005737  000001           TST     1
1529  005640  013700  177776   RETV:   MOV     CC,%0           ;GET THE "NEW" STATUS
1530  005644  022700  000357           CMP     #357,%0                 ;IS IT CORRECT
1531  005650  001405                   BEQ     1$              ;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                     ;ERROR! "NEW" STATUS IS INCORRECT
1536  005662  000240                   LOOP                    ;REPLACE THIS INSTRUCTION BY A
1537                                                            ;"JMP @R0" (000110) TO GET A
1538                                                            ;SCOPE LOOP AND HIT CONTINUE
1539  005664  005037  177776   1$:     CLR     CC
1540  005670  010700                   SCOPE
1541
```

```
                                        ;*************************************************************
                                        ;*TEST 52 TEST THAT AN ODDSOURCE INTERMEDIATE ADDRESS CAUSES AN ODD ADDRESS TRAP
                                        ;*************************************************************
005672  005237  000304          TST52:  INC     @#$TESTN
005676  012706  021774                  MOV     #BUFF,%6        ;SET UP STACK POINTER
005702  012737  005736  000004          MOV     #OAER0,4        ;LOAD ERROR VECTOR
005710  005037  000006                  CLR     6
005714  012701  000001                  MOV     #1,%1           ;LOAD INDEX REGISTER
005720  067100  000000                  ADD     @0(1),%0        ;SRC ADRS [0(1)] IS ODD

005724  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
005726  012713  000073                  MOV     #73,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
005732  000000                          HLT                     ;ERROR! ODD ADRES ERROR FAILED TO TRAP
005734  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
005736  010700                  OAER0:  SCOPE

005740  012706  021774                  MOV     #BUFF,%6
005744  012737  005774  000004          MOV     #OAER1,4
005752  012701  021773                  MOV     #BUFF-1,%1
005756  147100  000000                  BICB    @0(1),%0        ;SRC INT ADRS [0(1)] IS ODD

005762  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
005764  012713  000074                  MOV     #74,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
005770  000000                          HLT                     ;ERROR! ODD ADRS IN DST FAILED TO TRAP
005772  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
005774  010700                  OAER1:  SCOPE

                                        ;*************************************************************
                                        ;*TEST 53 TEST THAT AN ODD SOURCE FINAL ADDRESS WILL CAUSE AN ODD ADDRESS TRAP
                                        ;*************************************************************
005776  005237  000304          TST53:  INC     @#$TESTN
006002  012706  021774                  MOV     #BUFF,%6
006006  012737  006042  000004          MOV     #OAER2,4
006014  012737  000001  000100          MOV     #1,100
006022  005001                          CLR     %1
006024  017100  000100                  MOV     @100(1),%0      ;SRC FINAL ADRS IS ODD

006030  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
006032  012713  000075                  MOV     #75,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
006036  000000                          HLT                     ;ERROR! ODD FINAL SRC ADRS FAILED TO TRAP
006040  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
006042  010700                  OAER2:  SCOPE

                                        ;*************************************************************
                                        ;*TEST 54 TEST THAT AN ODD DEST INTERMEDIATE ADDRESS CAUSES AN ODD ADRS TRAP
                                        ;*************************************************************
006044  005237  000304          TST54:  INC     @#$TESTN
006050  012706  021774                  MOV     #BUFF,%6
006054  012737  006104  000004          MOV     #OAER3,4
006062  012701  000001                  MOV     #1,%1
```



```
006066  074071  000000                  XOR     %0,@0(1)        ;DST INT ADRS [0(1)] IS ODD

006072  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
006074  012713  000076                  MOV     #76,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
006100  000000                          HLT                     ;ERROR! ODD ADRS ERROR FAILED TO TRAP
006102  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
006104  010700                  OAER3:  SCOPE

006106  012706  021774                  MOV     #BUFF,%6
006112  012737  006136  000004          MOV     #OAER4,4
006120  005001                          CLR     %1
006122  122131                          CMPB    (1)+,@(1)+      ;DST INT ADRS IS ODD [(R1) +1]

006124  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
006126  012713  000077                  MOV     #77,(R3)        ;MOVE TO MAIL BOX ($FATAL) THE ERROR
006132  000000                          HLT                     ;ERROR! ODD ADRS ERROR FAILED TO TRAP
006134  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
006136  010700                  OAER4:  SCOPE

                                        ;*************************************************************
                                        ;*TEST 55 TEST THAT AN ODD DEST FINAL ADRS WILL CAUSE AN ODD ADRS TRAP
                                        ;*************************************************************
006140  005237  000304          TST55:  INC     @#$TESTN
006144  012706  021774                  MOV     #BUFF,%6
006150  012737  006203  000004          MOV     #OAER5,4
006156  012737  000001  000100          MOV     #1,100
006164  005001                          CLR     %1
006166  006771  000100                  SXT     @100(1)         ;DST FINAL ADRS IS ODD

006172  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
006174  012713  000100                  MOV     #100,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
006200  000000                          HLT                     ;ERROR! DST ODD ADRS ERR FAILED TO TRAP
006202  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
006204  010700                  OAER5:  SCOPE

                                        ;*************************************************************
                                        ;*TEST 56 TEST THAT CMPB @(R)+,@(R)+ DOES NOT FAIL
                                        ;*************************************************************
006206  005237  000304          TST56:  INC     @#$TESTN
006212  012706  021774                  MOV     #BUFF,%6        ;INITIALIZE THE STACK POINTER
006216  012737  006324  000004          MOV     #OAER6,4
006224  005001                          CLR     %1
006226  012737  021415  000000          MOV     #TEMP+1,0
006234  012737  021416  000002          MOV     #TEMP+2,2
006242  012737  123000  021414          MOV     #123000,TEMP
006250  012737  177246  021416          MOV     #177246,TEMP+2
006256  123131                          CMPB    @(1)+,@(1)+     ;COMP. LOC. (BYTE) TEMP+1 & TEMP+2
006260  001405                          BEQ     1$

006262  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
```

```
 1654  006264  012713  000101                  MOV     #101,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 1655  006270  000000                          HLT                     ;COMPARE INSTRUCTION FAILED
 1656  006272  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 1657                                                                  ;"JMP @R0" (000110) TO GET A
 1658                                                                  ;SCOPE LOOP AND HIT CONTINUE
 1659  006274  022701  000004          1$:     CMP     #4,R1           ;DID REGISTER INCREMENT PROPERLY
 1660  006300  001405                          BEQ     2$
 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                     ;ERROR! R1 DID NOT INC. BY +2 TWICE
 1665  006312  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 1666                                                                  ;"JMP @R0" (000110) TO GET A
 1667                                                                  ;SCOPE LOOP AND HIT CONTINUE
 1668  006314  000405                  2$:     BR      OAER8
 1669
 1670  006316  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 1671  006320  012713  000103                  MOV     #103,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 1672  006324  000000                  OAER6:  HLT                     ;ERROR! ODD ADDRESS ERROR
 1673  006326  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 1674                                                                  ;"JMP @R0" (000110) TO GET A
 1675                                                                  ;SCOPE LOOP AND HIT CONTINUE
 1676  006330  010700                  OAER8:  SCOPE
 1677
 1678                                          ;******************************************************************
 1679                                          ;*TEST 57 TEST THAT SWAB ODD ADDRESS CAUSES AN ODD ADDRESS TRAP
 1680                                          ;******************************************************************
 1681  006332  005237  000304          TST57:  INC     @#$TESTN
 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                     ;ERROR! FAILED TO TRAP
 1689  006364  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 1690                                                                  ;"JMP @R0" (000110) TO GET A
 1691                                                                  ;SCOPE LOOP AND HIT CONTINUE
 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                          SCOPE
 1697
```



```
 1698
 1699                                          ;******************************************************************
 1700                                          ;*TEST 60 TEST THAT SETTING THE 'T' BIT WILL CAUSE A TRAP TO 14
 1701                                          ;******************************************************************
 1702  006402  005237  000304          TST60:  INC     @#$TESTN
 1703  006406  012706  021774                  MOV     #BUFF,%6
 1704  006412  005001                          CLR     R1
 1705  006414  012737  006454  000014          MOV     #RETAT,TRTVEC   ;SET UP TO TRAP TO 14
 1706  006422  005037  000016                  CLR     TRTVEC+2
 1707  006426  012746  000020                  MOV     #20,-(6)                ;PUSH 'T'BIT ON THE STACK
 1708  006432  012746  006440                  MOV     #.+6,-(6)               ;PUSH PC ON THE STACK
 1709  006436  000002                          RTI                     ;SET 'T' BIT
 1710  006440  005201                          INC     R1              ;TRAP HERE
 1711
 1712  006442  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 1713  006444  012713  000105                  MOV     #105,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 1714  006450  000000                          HLT                     ;TRACE BIT DOES NOT TRAP
 1715  006452  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 1716                                                                  ;"JMP @R0" (000110) TO GET A
 1717                                                                  ;SCOPE LOOP AND HIT CONTINUE
 1718  006454  005701                  RETAT:  TST     R1              ;DID T BIT TRAP RIGHT AFTER RTI?
 1719  006456  001405                          BEQ     1$              ;YES
 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                     ;T BIT TRAP DID NOT OCCUR
 1724                                                                  ;RIGHT AFTER RTI (ONE INSTR
 1725                                                                  ;WAS ALLOWED AFTER RTI)
 1726  006470  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 1727                                                                  ;"JMP @R0" (000110) TO GET A
 1728                                                                  ;SCOPE LOOP AND HIT CONTINUE
 1729  006472  010700                  1$:     SCOPE
 1730                                          ;******************************************************************
 1731                                          ;*TEST 61 TEST STACK POINTER DECREMENTS ON A T BIT TRAP
 1732                                          ;******************************************************************
 1733  006474  005237  000304          TST61:  INC     @#$TESTN
 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                     ;STACK POINTER WAS NOT PUSHED TWICE BY T BIT TRAP
 1745  006542  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 1746                                                                  ;"JMP @R0" (000110) TO GET A
 1747                                                                  ;SCOPE LOOP AND HIT CONTINUE
 1748  006544  010700                  1$:     SCOPE
 1749                                          ;******************************************************************
 1750                                          ;*TEST 62 TEST FOR PROPER PC ON STACK ON A T BIT TRAP
 1751                                          ;******************************************************************
 1752  006546  005237  000304          TST62:  INC     @#$TESTN
 1753  006552  012706  021774                  MOV     #BUFF,%6
```

```
1754  006556  012737  006610  000014          MOV     #RETCT,TRTVEC
1755  006564  012746  000020                  MOV     #20,-(6)                ;PUSH 'T'BIT ON THE STACK
1756  006570  012746  006576                  MOV     #.+6,-(6)               ;PUSH PC ON THE STACK
1757  006574  000002                          RTI                             ;SET 'T' BIT
1758  006576                          RETCT3:
1759
1760  006576  005212                          INC     (P2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 61   006600  012713  000110                  MOV     #110,(R3)               ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1762  006604  000000                          HLT                             ;ERROR! 'T'BIT TRAP NOT ACKNOWLEDGED
1763                                                                          ;BEFORE 1ST INST, AFTER AN RTI
1764  006606  000240                          LOOP                            ;REPLACE THIS INSTRUCTION BY A
1765                                                                          ;"JMP @R0" (000110) TO GET A
1766                                                                          ;SCOPE LOOP AND HIT CONTINUE
1767  006610  022737  006576  021770  RETCT:  CMP     #RETCT3,BUFF-4
1768  006616  001405                          BEQ     1$
1769
1770  006620  005212                          INC     (R2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1771  006622  012713  000111                  MOV     #111,(R3)               ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1772  006626  000000                          HLT                             ;CORRECT PC WAS NOT SAVED ON STACK
1773                                                                          ;ON A T BIT TRAP
1774  006630  000240                          LOOP                            ;REPLACE THIS INSTRUCTION BY A
1775                                                                          ;"JMP @R0" (000110) TO GET A
1776                                                                          ;SCOPE LOOP AND HIT CONTINUE
1777  006632  022737  000020  021772  1$:     CMP     #20,BUFF-2              ;CORRECT OLD PSW SAVED ON STACK?
1778  006640  001405                          BEQ     2$
1779
1780  006642  005212                          INC     (P2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1781  006644  012713  000112                  MOV     #112,(R3)               ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1782  006650  000000                          HLT                             ;OLD PSW WAS NOT SAVED ON THE STACK
1783                                                                          ;CORRECTLY, ON A T BIT TRAP
1784  006652  000240                          LOOP                            ;REPLACE THIS INSTRUCTION BY A
1785                                                                          ;"JMP @R0" (000110) TO GET A
1786                                                                          ;SCOPE LOOP AND HIT CONTINUE
1787  006654  010700                  2$:     SCOPE
1788
1789
1790                                          ;*************************************************************
1791                                          ;*TEST 63 TEST FOR PROPER PC ON THE STACK, ON A T BIT TRAP (RTI)
1792                                          ;*************************************************************
1793  006656  005237  000304          TST63:  INC     @#STESTN
1794  006662  012706  021774                  MOV     #BUFF,SP                ;INITIALIZE THE STACK POINTER
1795  006666  012737  006724  000014          MOV     #RETCT1,TRTVEC          ;SET UP 'T' BIT TRAP VECTOR
1796  006674  005001                          CLR     R1
1797  006676  012746  000020                  MOV     #20,-(6)                ;PUSH 'T' BIT ON THE STACK
1798  006702  012746  006710                  MOV     #.+6,-(6)               ;PUSH PC ON THE STACK
1799  006706  000006                          RTT                             ;RETURN FROM INTERRUPT
1800  006710  005201                          INC     R1
1801                                  ;'T' BIT TRAP SHOULD OCCUR HERE
1802  006712                          RETCT2:
1803
1804  006712  005212                          INC     (P2)                    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1805  006714  012713  000113                  MOV     #113,(R3)               ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1806  006720  000000                          HLT                             ;ERROR!'T' BIT DID NOT TRAP
1807  006722  000240                          LOOP                            ;REPLACE THIS INSTRUCTION BY A
1808                                                                          ;"JMP @R0" (000110) TO GET A
1809                                                                          ;SCOPE LOOP AND HIT CONTINUE
```



```
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 @R0" (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 @R0" (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  005737  000016                  CLR     TPTVEC+2                ;TO HALT AT 16
1836  006776  010700                          SCOPE
```

```
1837
1838                                            ;*****************************************************************
1839                                            ;*TEST 64 TEST NO STACK OVERFLOW TRAP ON A DATI TO STACK LOC. LESS THAN 400
1840                                            ;*****************************************************************
1841  007400  005237  000304        TST64:  INC     @#STESTN
1842  007404  012706  000376                MOV     #376,%6         ;SET STACK POINTER LESS THAN 400
1843  007410  012737  007434  000004        MOV     #TDEC1,ERRVEC   ;LOAD TRAP VECTOR
1844  007416  005037  000006                CLR     ERRVEC+2
1845  007422  005716                        TST     (6)             ;DATI FROM LOC 376
1846  007424  000405                        BR      TDEC1A
1847
1848  007426  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1849  007430  012713  000116                MOV     #116,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1850  007434  000000                TDEC1:  HLT                     ;ERROR! STACK OVERFLOW TRAP OCCURED
1851                                                                ;ON A DATI TO LOC LESS THAN 400
1852  007436  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
1853                                                                ;"JMP @R0" (000110) TO GET A
1854                                                                ;SCOPE LOOP AND HIT CONTINUE
1855  007440  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  007442  005237  000304        TST65:  INC     @#STESTN
1861  007446  012706  000376                MOV     #376,%6
1862  007452  012737  007476  000004        MOV     #TDEC2,ERRVEC
1863  007460  005016                        CLR     (6)             ;DATIP/DATO TO BE 376
1864  007462  000240                        NOP
1865
1866  007464  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1867  007466  012713  000117                MOV     #117,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1868  007472  000000                        HLT                     ;ERROR! NO STACK OVERFLOW TRAP (YELLOW)
1869  007474  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
1870                                                                ;"JMP @R0" (000110) TO GET A
1871                                                                ;SCOPE LOOP AND HIT CONTINUE
1872  007476  010700                TDEC2:  SCOPE
1873
1874                                            ;*****************************************************************
1875                                            ;*TEST 66 TEST THAT A DATIP/DATOB CAUSES AN OVERFLOW TRAP
1876                                            ;*****************************************************************
1877  007500  005237  000304        TST66:  INC     @#STESTN
1878  007504  012706  000376                MOV     #376,%6
1879  007510  012737  007534  000004        MOV     #TDEC2A,ERRVEC
1880  007516  152716  177777                BISB    #-1,(6)
1881
1882  007522  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1883  007524  012713  000120                MOV     #120,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1884  007530  000000                        HLT                     ;ERROR! NO STACK OVERFLOW TRAP (YELLOW ZONE)
1885  007532  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
1886                                                                ;"JMP @R0" (000110) TO GET A
1887                                                                ;SCOPE LOOP AND HIT CONTINUE
1888  007534  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  007536  005237  000304        TST67:  INC     @#STESTN
1894  007542  012705  001000                MOV     #1000,%5
1895  007546  012706  000376                MOV     #376,%6
1896  007552  012737  007572  000004        MOV     #TDEC3,ERRVEC
1897  007560  124645                        CMPB    -(6),-(5)
1898  007562  000405                        BR      TDEC6
1899
1900  007564  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1901  007566  012713  000121                MOV     #121,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1902  007572  000000                TDEC3:  HLT                     ;ERROR! STACK OVERFLOW TRAP OCCURED
1903                                                                ;ON A DATI TO STACK LOC UNDER 400
1904  007574  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
1905                                                                ;"JMP @R0" (000110) TO GET A
1906                                                                ;SCOPE LOOP AND HIT CONTINUE
1907  007576  010700                TDEC6:  SCOPE
1908
1909  007600  012706  000400                MOV     #400,%6
1910  007604  012737  007624  000004        MOV     #TDEC4,ERRVEC
1911  007612  134546                        BITB    -(5),-(6)
1912  007614  000405                        BR      TDEC6A
1913
1914  007616  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1915  007620  012713  000122                MOV     #122,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1916  007624  000000                TDEC4:  HLT                     ;ERROR! STACK OVERFLOW TRAP OCCURRED
1917                                                                ;ON A DATI TO STACK LOC UNDER 400
1918  007626  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
1919                                                                ;"JMP @R0" (000110) TO GET A
1920                                                                ;SCOPE LOOP AND HIT CONTINUE
1921  007630  010700                TDEC6A: SCOPE
1922
1923                                            ;*****************************************************************
1924                                            ;*TEST 70 TEST THAT OVERFLOW TRAP DOES NOT LOSE INFORMATION (OLD PC & PS)
1925                                            ;*****************************************************************
1926  007632  005237  000304        TST70:  INC     @#STESTN
1927  007636  012706  000400                MOV     #400,%6
1928  007642  005037  000376                CLR     376             ;STATUS WORD OF LOC 10
1929  007646  005037  000374                CLR     374
1930  007652  012737  007304  000004        MOV     #TDEC5,ERRVEC   ;RETURN TO LOC 4
1931  007660  012737  000017  177776        MOV     #17,CC          ;PRE SET STATUS
1932  007666  005246                        INC     -(6)
1933
1934  007670  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1935  007672  012713  000123                MOV     #123,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1936  007676  000000                TDEC5A: HLT                     ;ERROR! NO STACK OVERFLOW TRAP (YELLOW)
1937  007700  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
1938                                                                ;"JMP @R0" (000110) TO GET A
1939                                                                ;SCOPE LOOP AND HIT CONTINUE
1940  007702  000415                        BR      TDEC5B          ;GO TO SCOPE
1941  007704  022737  000001  000376 TDEC5:  CMP     #1,376          ;WAS INC   -(6) EXECUTED
1942  007712  001405                        BEQ     1$
1943
1944  007714  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
1945  007716  012713  000124                MOV     #124,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
1946  007722  000000                        HLT                     ;ERROR! INSTRUCTION CAUSING YELLOW
1947                                                                ;ZONE VIOLATION WAS ABORTED, IT
1948                                                                ;SHOULD HAVE BEEN EXECUTED BEFORE
```

```
                                        T70 TEST THAT OVERFLOW TRAP DOES NOT LOSE INFORMATION (OLD PC & PS)

                                                                        ;TRAPPING
007324  000240                                LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                        ;"JMP @R0" (000110) TO GET A
                                                                        ;SCOPE LOOP AND HIT CONTINUE
007326  022737  000001  000374  10:      CMP     #1,174                ;WAS STATUS SAVED NOTE: INC DOES NOT
                                                                        ;AFFECT 'C' BIT IN STATUS.
007334  001400                          BEQ     TDEC5R
007336  010700                  TDEC5R: SCOPE


                                        ;*************************************************************
                                        ;*TEST 71 TEST THAT A RESERVED  INST CAUSES AN OVERFLOW TRAP
                                        ;*************************************************************
007340  005237  000304          TST71:  INC     @#$TESTN
007344  012706  000400                  MOV     #400,%6               ;SET UP STACK TO OVERFLOW
007350  012737  007404  000010          MOV     #VDEC2,10             ;SET UP INST VECTOR
007356  012737  007410  000004          MOV     #VDEC1,4              ;SET UP OVERFLOW VECTOR
007364  075040                          75040                         ;THIS TRAP SHOULD CAUSE OVERFLOW

007366  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007370  012713  000125                  MOV     #125,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007374  000000                          HLT                           ;NO TRAP OCCURRED

007376  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007400  012713  000126                  MOV     #126,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007404  000000                  VDEC2:  HLT                           ;TRAP FLAG OVERFLOW DID NOT OCCUR
007406  000240                          LOOP                          ;REPLACE THIS INSTRUCTION BY A
                                                                      ;"JMP @R0" (000110) TO GET A
                                                                      ;SCOPE LOOP AND HIT CONTINUE
007410  010700                  VDEC1:  SCOPE                         ;NORMAL OVERFLOW RETURN


                                        ;*************************************************************
                                        ;*TEST 72 TEST THAT AN "IOT" CAUSES AN OVERFLOW TRAP
                                        ;*************************************************************
007412  005237  000304          TST72:  INC     @#$TESTN
007416  012706  000400                  MOV     #400,%6               ;SET UP STACK TO OVERFLOW
007422  012737  007456  000020          MOV     #VDEC4,20             ;SET UP INST VECTOR
007430  012737  007462  000004          MOV     #VDEC3,4              ;SET UP OVERFLOW VECTOR
007436  000004                          IOT                           ;THIS TRAP SHOULD CAUSE OVERFLOW

007440  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007442  012713  000127                  MOV     #127,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007446  000000                          HLT                           ;NO TRAP OCCURRED

007450  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007452  012713  000130                  MOV     #130,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007456  000000                  VDEC4:  HLT                           ;TRAP FLAG OVERFLOW DID NOT OCCUR
007460  000240                          LOOP                          ;REPLACE THIS INSTRUCTION BY A
                                                                      ;"JMP @R0" (000110) TO GET A
                                                                      ;SCOPE LOOP AND HIT CONTINUE
007462  010700                  VDEC3:  SCOPE                         ;NORMAL OVERFLOW RETURN
                                        ;*************************************************************
                                        ;*TEST 73 TEST THAT AN "EMT" CAUSES AN OVERFLOW TRAP
                                        ;*************************************************************
007464  005237  000304          TST73:  INC     @#$TESTN
007470  012706  000400                  MOV     #400,%6               ;SET UP STACK TO OVERFLOW
```



```
                                        T73 TEST THAT AN "EMT" CAUSES AN OVERFLOW TRAP

007474  012737  007530  000030          MOV     #VDEC6,30             ;SET UP INST VECTOR
007502  012737  007534  000004          MOV     #VDEC5,4              ;SET UP OVERFLOW VECTOR
007510  104000                          EMT                           ;THIS TRAP SHOULD CAUSE OVERFLOW

007512  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007514  012713  000131                  MOV     #131,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007520  000000                          HLT                           ;NO TRAP OCCURRED

007522  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007524  012713  000132                  MOV     #132,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007530  000000                  VDEC6:  HLT                           ;TRAP FLAG OVERFLOW DID NOT OCCUR
007532  000240                          LOOP                          ;REPLACE THIS INSTRUCTION BY A
                                                                      ;"JMP @R0" (000110) TO GET A
                                                                      ;SCOPE LOOP AND HIT CONTINUE
007534  010700                  VDEC5:  SCOPE                         ;NORMAL OVERFLOW RETURN


                                        ;*************************************************************
                                        ;*TEST 74 TEST THAT AN "TRAP" CAUSES AN OVERFLOW TRAP
                                        ;*************************************************************
007536  005237  000304          TST74:  INC     @#$TESTN
007542  012706  000400                  MOV     #400,%6               ;SET UP STACK TO OVERFLOW
007546  012737  007602  000034          MOV     #VDEC8,34             ;SET UP INST VECTOR
007554  012737  007606  000004          MOV     #VDEC7,4              ;SET UP OVERFLOW VECTOR
007562  104400                          TRAP                          ;THIS TRAP SHOULD CAUSE OVERFLOW

007564  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007566  012713  000133                  MOV     #133,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007572  000000                          HLT                           ;NO TRAP OCCURRED

007574  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007576  012713  000134                  MOV     #134,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007602  000000                  VDEC8:HLT                             ;TRAP FLAG OVERFLOW DID NOT OCCUR
007604  000240                          LOOP                          ;REPLACE THIS INSTRUCTION BY A
                                                                      ;"JMP @R0" (000110) TO GET A
                                                                      ;SCOPE LOOP AND HIT CONTINUE
007606  010700                  VDEC7:SCOPE                           ;NORMAL OVERFLOW RETURN


                                        ;*************************************************************
                                        ;*TEST 75 TEST THAT AN BPT CAUSES AN OVERFLOW TRAP
                                        ;*************************************************************
007610  005237  000304          TST75:  INC     @#$TESTN
007614  012706  000400                  MOV     #400,%6               ;SET UP STACK TO OVERFLOW
007620  012737  007654  000014          MOV     #VDEC10,14            ;SET UP INST VECTOR
007626  012737  007660  000004          MOV     #VDEC9,4              ;SET UP OVERFLOW VECTOR
007634  000003                          BPT                           ;THIS TRAP SHOULD CAUSE OVERFLOW

007636  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007640  012713  000135                  MOV     #135,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007644  000000                          HLT                           ;NO TRAP OCCURRED

007646  005212                          INC     (R2)                  ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
007650  012713  000136                  MOV     #136,(R3)             ;MOVE TO MAIL BOX ($FATAL) THE ERROR
007654  000000                  VDEC10: HLT                           ;TRAP FLAG OVERFLOW DID NOT OCCUR
007656  000240                          LOOP                          ;REPLACE THIS INSTRUCTION BY A
                                                                      ;"JMP @R0" (000110) TO GET A
                                                                      ;SCOPE LOOP AND HIT CONTINUE
```

```
 2061  007660  010700                  VDEC9:  SCOPE                   ;NORMAL OVERFLOW RETURN
 2062
 2063                                          ;*************************************************************
 2064                                          ;*TEST 76 TEST THAT AN ILLEGAL INSTR. CAUSES AN OVERFLOW TRAP
 2065                                          ;*************************************************************
 2066  007662  005237  000304          TST76:  INC     @#$TESTN
 2067  007666  012706  000400                  MOV     #400,%6         ;SET UP STACK TO OVERFLOW
 2068  007672  012737  007726  000004          MOV     #VDEC12,4       ;SET UP INST VECTOR
 2069  007700  012737  007732  000004          MOV     #VDEC11,4       ;SET UP OVERFLOW VECTOR
 2070  007706  004700                          ILLA                    ;THIS TRAP SHOULD CAUSE OVERFLOW
 2071
 2072  007710  005212                          INC     (R2)            ;SET MESSAGE TYPE ($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 ($FATAL) THE ERROR
 2078  007726  000000                  VDEC12: HLT                     ;TRAP FLAG OVERFLOW DID NOT OCCUR
 2079  007730  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2080                                                                  ;"JMP @R0" (000110) TO GET A
 2081                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2082  007732  010700                  VDEC11: SCOPE                   ;NORMAL OVERFLOW RETURN
 2083  007734  020627  000370                  CMP     %6,#370         ;STACK PUSHED FOUR WORDS?
 2084  007740  001405                          BEQ     1$
 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 @R0" (000110) TO GET A
 2091                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2092  007754  010700                  1$:     SCOPE
 2093
 2094                                          ;*************************************************************
 2095                                          ;*TEST 77 TEST THAT AN ILLEGAL INSTR. CAUSES AN OVERFLOW TRAP
 2096                                          ;*************************************************************
 2097  007756  005237  000304          TST77:  INC     @#$TESTN
 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                          ILLA                    ;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 @R0" (000110) TO GET A
 2112                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2113  010026  010700                  VDEC13: SCOPE                   ;NORMAL OVERFLOW RETURN
 2114
 2115
 2116                                          ;INSTRUCTION EQUATE STATEMENTS
```



```
 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     @#$TESTN
 2127  010034  012706  000400                  MOV     #400,%6 ;SET STACK POINTER
 2128  010040  005000                          CLR     %0              ;PRE SET R0
 2129  010042  012737  010064  000004          MOV     #VDEC15,4       ;LOAD ERROR VECTOR
 2130  010050  004510                          4510                    ;CAUSE OVERFLOW
 2131
 2132  010052  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2133  010054  012713  000144                  MOV     #144,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2134  010060  000000                          HLT                     ;ERROR! OVERFLOW FAILED TO TRAP
 2135  010062  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY 15
 2136                                                                  ;"JMP @R0" (000110) TO GET 15
 2137                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2138  010064  022706  000372          VDEC15: CMP     #372,%6         ;HAS STACK POINTER MOVED BY 6
 2139  010070  001405                          BEQ     1$              ;(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 @R0" (000110) TO GET 15
 2149                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2150  010104  010700                  1$:     SCOPE
 2151
 2152                                          ;*************************************************************
 2153                                          ;*TEST 101 TEST THAT THE INSTRUCTION (005046) CAUSES STACK OVERFLOW CONDITION
 2154                                          ;*************************************************************
 2155  010106  005237  000304          TST101: INC     @#$TESTN
 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 @R0" (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)
```

```
                                                        ;R6 DID NOT DECREMENT BY 6 WHEN
                                                        ;AN INSTRUCTION DOING AUTO-DECREMENT
                                                        ;OF R6 CAUSED YELLOW ZONE VIOLATION
010160  000240                    LOOP                ;REPLACE THIS INSTRUCTION BY 16
                                                        ;"JMP @R0" (000110) TO GET 16
                                                        ;SCOPE LOOP AND HIT CONTINUE
010162  010700            1$:     SCOPE

                                ;*************************************************************
                                ;*TEST 102 TEST THAT THE INSTRUCTION (010046) CAUSES STACK OVERFLOW CONDITION
                                ;*************************************************************
010164  005237  000304    TST102: INC     @#$TESTN
010170  012706  000400            MOV     #400,%6 ;SET STACK POINTER
010174  005000                    CLR     %0              ;PRE SET R0
010176  012737  010220  000004    MOV     #VDEC17,4       ;LOAD ERROR VECTOR
010204  010046                    010046                  ;CAUSE OVERFLOW

010206  005212                    INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010210  012713  000150            MOV     #150,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010214  000000                    HLT                     ;ERROR! OVERFLOW FAILED TO TRAP
010216  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY 17
                                                        ;"JMP @R0" (000110) TO GET 17
                                                        ;SCOPE LOOP AND HIT CONTINUE
010220  022706  000372    VDEC17: CMP     #372,%6         ;HAS STACK POINTER MOVED BY 6
010224  001405                    BEQ     1$              ;(2 FOR THE AUTO DECREMENT + 4 FOR

010226  005212                    INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010230  012713  000151            MOV     #151,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010234  000000                    HLT                     ;THE ERROR TRAP)
                                                        ;R6 DID NOT DECREMENT BY 6 WHEN
                                                        ;AN INSTRUCTION DOING AUTO-DECREMENT
                                                        ;OF R6 CAUSED YELLOW ZONE VIOLATION
010236  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY 17
                                                        ;"JMP @R0" (000110) TO GET 17
                                                        ;SCOPE LOOP AND HIT CONTINUE
010240  010700            1$:     SCOPE

                                ;*************************************************************
                                ;*TEST 103 TEST THAT THE INSTRUCTION (006746) CAUSES STACK OVERFLOW CONDITION
                                ;*************************************************************
010242  005237  000304    TST103: INC     @#$TESTN
010246  012706  000400            MOV     #400,%6 ;SET STACK POINTER
010252  005000                    CLR     %0              ;PRE SET R0
010254  012737  010276  000004    MOV     #VDEC18,4       ;LOAD ERROR VECTOR
010262  006746                    006746                  ;CAUSE OVERFLOW

010264  005212                    INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010266  012713  000152            MOV     #152,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010272  000000                    HLT                     ;ERROR! OVERFLOW FAILED TO TRAP
010274  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY 18
                                                        ;"JMP @R0" (000110) TO GET 18
                                                        ;SCOPE LOOP AND HIT CONTINUE
010276  022706  000372    VDEC18: CMP     #372,%6         ;HAS STACK POINTER MOVED BY 6
010302  001405                    BEQ     1$              ;(2 FOR THE AUTO DECREMENT + 4 FOR

010304  005212                    INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
```



```
010306  012713  000153                    MOV     #153,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010312  000000                            HLT                     ;THE ERROR TRAP)
                                                                ;R6 DID NOT DECREMENT BY 6 WHEN
                                                                ;AN INSTRUCTION DOING AUTO-DECREMENT
                                                                ;OF R6 CAUSED YELLOW ZONE VIOLATION
010314  000240                            LOOP                    ;REPLACE THIS INSTRUCTION BY 18
                                                                ;"JMP @R0" (000110) TO GET 18
                                                                ;SCOPE LOOP AND HIT CONTINUE
010316  010700                    1$:     SCOPE

                                        ;*************************************************************
                                        ;*TEST 104 TEST THAT THE INSTRUCTION (074046) CAUSES STACK OVERFLOW CONDITION
                                        ;*************************************************************
010320  005237  000304            TST104: INC     @#$TESTN
010324  012706  000400                    MOV     #400,%6 ;SET STACK POINTER
010330  005000                            CLR     %0              ;PRE SET R0
010332  012737  010354  000004            MOV     #VDEC19,4       ;LOAD ERROR VECTOR
010340  074046                            074046                  ;CAUSE OVERFLOW

010342  005212                            INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010344  012713  000154                    MOV     #154,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010350  000000                            HLT                     ;ERROR! OVERFLOW FAILED TO TRAP
010352  000240                            LOOP                    ;REPLACE THIS INSTRUCTION BY 19
                                                                ;"JMP @R0" (000110) TO GET 19
                                                                ;SCOPE LOOP AND HIT CONTINUE
010354  022706  000372            VDEC19: CMP     #372,%6         ;HAS STACK POINTER MOVED BY 6
010360  001405                            BEQ     1$              ;(2 FOR THE AUTO DECREMENT + 4 FOR

010362  005212                            INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010364  012713  000155                    MOV     #155,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010370  000000                            HLT                     ;THE ERROR TRAP)
                                                                ;R6 DID NOT DECREMENT BY 6 WHEN
                                                                ;AN INSTRUCTION DOING AUTO-DECREMENT
                                                                ;OF R6 CAUSED YELLOW ZONE VIOLATION
010372  000240                            LOOP                    ;REPLACE THIS INSTRUCTION BY 19
                                                                ;"JMP @R0" (000110) TO GET 19
                                                                ;SCOPE LOOP AND HIT CONTINUE
010374  010700                    1$:     SCOPE
```

```
                                        ;*************************************************************
                                        ;*TEST 105 TEST FOR FALSE OVERFLOW TRAP
                                        ;*************************************************************
010376  005237  000304        TST105: INC     @#$TESTN
                              ;PROGRAM MAY HAVE RELOADED IF OVERFLOW FAILS
010402  012737  010500  000004        MOV     #FOVFP,4        ;SET UP OVERFLOW POINTER
010410  012706  000402                MOV     #402,%6
010414  010605                        MOV     R6,R5
010416  005746                        TST     -(6)            ;SHOULD NOT OVERFLOW
010420  012706  001002                MOV     #1002,%6
010424  010605                        MOV     R6,R5
010426  005746                        TST     -(6)            ;SHOULD NOT OVERFLOW
010430  012706  002002                MOV     #2002,%6
010434  010605                        MOV     R6,R5
010436  005746                        TST     -(6)            ;SHOULD NOT OVERFLOW
010440  012706  004002                MOV     #4002,%6
010444  010605                        MOV     R6,R5
010446  005746                        TST     -(6)            ;SHOULD NOT OVERFLOW
010450  012706  010002                MOV     #10002,%6
010454  010605                        MOV     R6,R5
010456  005746                        TST     -(6)
010460  012706  020000                MOV     #20000,%6       ;SHOULD NOT OVERFLOW
010464  010605                        MOV     R6,R5
010466  005746                        TST     -(6)
010470  000105                        RP      FOV1
                                        ;
010472  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010474  012713  000156                MOV     #156,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010500  000000        FOVER:  HLT             ;FALSE STACK OVERFLOW TRAP OCCURRED
                                        ;ON DOING AN AUTO-DECREMENT OF R6
                                        ;THE INITIAL R6 IS IN SAVED IN R5
                                        ;CHECK STACK TO FIND WHERE
010502  000240                LOOP            ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
010504  010700        FOV1:   SCOPE           ;SCOPE LOOP AND HIT CONTINUE
                                        ;*************************************************************
                                        ;*TEST 106 TEST FOR REPEATED TRAPS AND OVERFLOW OF STACK
                                        ;*************************************************************
010506  005237  000304        TST106: INC     @#$TESTN
                              ;WHAT THE STACK LOOKS LIKE WHEN INSTRUCTION AT "04" IS BEING EXECUTED
                              ;       OVFLW   PC=05,STAT=0    04      342  BOTTOM OF STACK
                              ;                               4       344
                              ;       TRT     PC=04,STAT=4    03A     346
                              ;                               0       350
                              ;       OVFLW   PC=05,STAT=0    03      352
                              ;                               3       354
                              ;       IOT     PC=03,STAT=3    02A     356
                              ;                               0       360
                              ;       OVFLW   PC=05,STAT=0    02      362
                              ;                               2       364
                              ;       TRAP    PC=02,STAT=2    01A     366
                              ;                               0       370
```



```
                              ;       OVFLW   PC=05,STAT=0    01      372
                              ;                               1       374
                              ;       EMT     PC=01,STAT=1    00A     376
                              ;                               17      400
                              ;                               402 INITIAL STACK POINTER
010512  012706  000402                MOV     #402,%6         ;INITIALIZE STACK POINTER
010516  012737  010632  000030        MOV     #01,EMTVEC      ;NEW PC POINTS TO NEXT INSTR.
010524  012737  000001  000032        MOV     #1,EMTVEC+2     ;NEW PSW
010532  012737  010646  000034        MOV     #02,TRAPVEC     ;NEW PC POINTS TO NEXT INSTR
010540  012737  000002  000036        MOV     #2,TRAPVEC+2    ;NEW PSW
010546  012737  010662  000020        MOV     #03,IOTVEC      ;NEW PC POINTS TO NEXT INSTR
010554  012737  000003  000022        MOV     #3,IOTVEC+2     ;NEW PSW
010562  012737  010676  000014        MOV     #04,TRTVEC      ;NEW PC POINTS TO NEXT INSTR
010570  012737  000004  000016        MOV     #4,TRTVEC+2     ;NEW PSW
010576  012737  011324  000004        MOV     #05,ERRVEC      ;NEW PC POINTS TO NEXT INSTR
010604  005037  000006                CLR     ERRVEC+2        ;NEW PSW
010610  012737  000017  177776        MOV     #17,CC          ;PRESENT PSW
010616  104000        0A:     EMT
010622          0A1:
                                        ;
010620  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010622  012713  000157                MOV     #157,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010626  000000        0AA:    HLT             ;EMT NOT EXECUTED
010630  000240                LOOP            ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
                                        ;SCOPE LOOP AND HIT CONTINUE
010632  104400        01:     TRAP            ;RETURN HERE FROM PREVIOUS YELLOW ZONE
                                        ;STACK VIOLATION
010634          011:
                                        ;
010634  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010636  012713  000160                MOV     #160,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010642  000000        01A:    HLT             ;TRAP NOT EXECUTED
010644  000240                LOOP            ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
                                        ;SCOPE LOOP AND HIT CONTINUE
010646  000004        02:     IOT             ;RETURN HERE FROM PREVIOUS YELLOW ZONE
                                        ;STACK VIOLATION
010650          021:
                                        ;
010650  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010652  012713  000161                MOV     #161,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010656  000000        02A:    HLT             ;IOT NOT EXECUTED
010660  000240                LOOP            ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
                                        ;SCOPE LOOP AND HIT CONTINUE
010662  000003        03:     BPT             ;RETURN HERE FROM PREVOUS YELLOW ZONE
                                        ;STACK VIOLATION
010664          031:
                                        ;
010664  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
010666  012713  000162                MOV     #162,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
010672  000000        03A:    HLT             ;BPT NOT EXECUTED
010674  000240                LOOP            ;REPLACE THIS INSTRUCTION BY A
                                        ;"JMP @R0" (000110) TO GET A
                                        ;SCOPE LOOP AND HIT CONTINUE
```

```
 2381  010676  022706  000342          04:     CMP     #342,SP         ;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                     ;ERROR1 INCORRECT STACK POINTER AFTER
 2387                                                                  ;4 REPETITIVE STACK OVERFLOW TRAPS
 2388  010714  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2389                                                                  ;"JMP @R0" (000110) TO GET A
 2390                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2391  010716  012700  000402          20$:    MOV     #402,R0
 2392  010722  022740  000017                  CMP     #17,-(R0)       ;CORRECT OLD PSW SAVED?
 2393  010726  001405                          BEQ     1$
 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 @R0" (000110) TO GET A
 2401                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2402  010742  022740  010620          1$:     CMP     #001,-(R0)
 2403  010746  001405                          BEQ     2$
 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 @R0" (000110) TO GET A
 2411                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2412  010762  022740  000001          2$:     CMP     #1,-(R0)
 2413  010766  001405                          BEQ     3$
 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 @R0" (000110) TO GET A
 2421                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2422  011002  022740  010632          3$:     CMP     #01,-(R0)
 2423  011006  001405                          BEQ     4$
 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 @R0" (000110) TO GET A
 2431                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2432  011022  022740  000000          4$:     CMP     #0,-(R0)
 2433  011026  001405                          BEQ     5$
 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 @R0" (000110) TO GET A
 2441                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2442  011042  022740  010634          5$:     CMP     #011,-(R0)
 2443  011046  001405                          BEQ     6$
 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 @R0" (000110) TO GET A
 2451                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2452  011062  022740  000002          6$:     CMP     #2,-(R0)
 2453  011066  001405                          BEQ     7$
 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 @R0" (000110) TO GET A
 2461                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2462  011102  022740  010646          7$:     CMP     #02,-(R0)
 2463  011106  001405                          BEQ     8$
 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 @R0" (000110) TO GET A
 2471                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2472  011122  022740  000000          8$:     CMP     #0,-(R0)
 2473  011126  001405                          BEQ     9$
 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 @R0" (000110) TO GET A
 2481                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2482  011142  022740  010650          9$:     CMP     #021,-(R0)
 2483  011146  001405                          BEQ     10$
 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 @R0" (000110) TO GET A
 2491                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2492  011162  022740  000003          10$:    CMP     #3,-(R0)
```

```
 2493  011166  000405                          BEQ     11$
 2494
 2495  011170  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2496  011172  012713  000176                  MOV     #176,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2497  011176  000000                          HLT                     ;CORRECT PSW WAS NOT SAVED ON STACK
 2498                                                                  ;OVERFLOW FOLLOWING IOT
 2499  011200  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2500                                                                  ;"JMP @R0" (000110) TO GET A
 2501                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2502  011202  022740  010662          11$:    CMP     #03,-(0)
 2503  011206  001405                          BEQ     12$
 2504
 2505  011210  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2506  011212  012713  000177                  MOV     #177,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2507  011216  000000                          HLT                     ;CORRECT PC WAS NOT SAVED ON STACK
 2508                                                                  ;OVERFLOW FOLLOWING IOT
 2509  011220  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2510                                                                  ;"JMP @R0" (000110) TO GET A
 2511                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2512  011222  022740  000000          12$:    CMP     #0,-(0)
 2513  011226  001405                          BEQ     13$
 2514
 2515  011230  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2516  011232  012713  000200                  MOV     #200,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2517  011236  000000                          HLT                     ;CORRECT PSW WAS NOT SAVED
 2518                                                                  ;ON BPT TRAP
 2519  011240  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2520                                                                  ;"JMP @R0" (000110) TO GET A
 2521                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2522  011242  022740  010664          13$:    CMP     #031,-(0)
 2523  011246  001405                          BEQ     14$
 2524
 2525  011250  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2526  011252  012713  000201                  MOV     #201,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2527  011256  000000                          HLT                     ;CORRECT PC WAS NOT SAVED
 2528                                                                  ;ON BPT TRAP
 2529  011260  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2530                                                                  ;"JMP @R0" (000110) TO GET A
 2531                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2532  011262  022740  000004          14$:    CMP     #4,-(0)
 2533  011266  001405                          BEQ     15$
 2534
 2535  011270  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2536  011272  012713  000202                  MOV     #202,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2537  011276  000000                          HLT                     ;CORRECT PSW WAS NOT SAVED ON STACK
 2538                                                                  ;OVERFLOW FOLLOWING BPT
 2539  011300  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2540                                                                  ;"JMP @R0" (000110) TO GET A
 2541                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2542  011302  022740  010676          15$:    CMP     #04,-(0)
 2543  011306  001405                          BEQ     07
 2544
 2545  011310  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2546  011312  012713  000203                  MOV     #203,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2547  011316  000000                  STKERR: HLT                     ;CORRECT PC WAS NOT SAVED ON STACK
 2548                                                                  ;OVERFLOW FOLLOWING BPT
```



```
 2549
 2550  011320  000240                          LOOP                    ;R0 GIVES LOCATION OF STACK ERROR
 2551                                                                  ;REPLACE THIS INSTRUCTION BY A
 2552                                                                  ;"JMP @R0" (000110) TO GET A
 2553  011322  000402                  07:     BR      06              ;SCOPE LOOP AND HIT CONTINUE
 2554  011324  000176  000000          05:     JMP     @(6)            ;RETURN FROM STACK
 2555  011330  010700                  06:     SCOPE                   ;OVERFLOW TRAP
 2556
 2557                                          ;*************************************************************
 2558                                          ;*TEST 107 TEST TRACE TRAP WHEN STACK IS IN YELLOW AREA
 2559                                          ;*************************************************************
 2560  011332  005237  000304          TST107: INC     @#$TESTN
 2561  011336  005037  021414                  CLR     TEMP
 2562  011342  012706  000400                  MOV     #400,%6         ;SET UP STACK POINTER
 2563  011346  012737  011532  000014          MOV     #TRTC,TRTVEC    ;LOAD TRT VECTOR
 2564  011354  005037  000016                  CLR     TRTVEC+2
 2565  011360  012737  011416  000004          MOV     #TRTB,ERRVEC    ;LOAD OVFLW. ERROR VECTOR
 2566  011366  005037  000006                  CLR     ERRVEC+2
 2567  011372  013737  011410  021416          MOV     TRTA,TEMP+2     ;SAVE FIRST WORD OF INST. AT TRTA
 2568  011400  012737  000003  011410          MOV     #BPT,TRTA       ;REPLACE WITH TRT INSTRUCTION
 2569  011406  000277                          SCC                     ;SET CONDITION CODES
 2570  011410  005137  021414          TRTA:   COM     TEMP            ;NOTE: THIS INSTRUCTION WAS REPLACED BY TRT
 2571  011414  000535                  TRTAA:  BR      TRTF
 2572  011416  022737  000017  000376  TRTB:   CMP     #17,376         ;WERE CONDITION CODES SAVED
 2573  011424  001405                          BEQ     1$
 2574
 2575  011426  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2576  011430  012713  000204                  MOV     #204,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2577  011434  000000                          HLT                     ;ERROR! CONDITION CODES NOT SAVED
 2578  011436  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2579                                                                  ;"JMP @R0" (000110) TO GET A
 2580                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2581  011440  022737  011412  000374  1$:     CMP     #TRTA+2,374     ;WAS RETURN FROM TRT SAVED
 2582  011446  001406                          BEQ     2$
 2583
 2584  011450  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2585  011452  012713  000205                  MOV     #205,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2586  011456  000000                          HLT                     ;ERROR! RETURN PC NOT SAVED
 2587  011460  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2588                                                                  ;"JMP @R0" (000110) TO GET A
 2589                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2590  011462  000523                          BR      TRTEX           ;EXIT TEST
 2591  011464  005737  000372          2$:     TST     372             ;WAS TRT STATUS SAVED?
 2592  011470  001405                          BEQ     3$
 2593
 2594  011472  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2595  011474  012713  000206                  MOV     #206,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2596  011500  000000                          HLT                     ;ERROR! TRT STATUS (TRTVEC+2) NOT SAVED
 2597  011502  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2598                                                                  ;"JMP @R0" (000110) TO GET A
 2599                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2600  011504  022737  011532  000370  3$:     CMP     #TRTC,370       ;TRT VECTOR SAVED
 2601  011512  001406                          BEQ     4$
 2602
 2603  011514  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2604  011516  012713  000207                  MOV     #207,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
```

```
 2605  011522  000000                        HLT                     ;ERROR! TRT VECTOR (TRTVEC) NOT SAVED
 2606  011524  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2607                                                                 ;"JMP @R0" (000110) TO GET A
 2608                                                                 ;SCOPE LOOP AND HIT CONTINUE
 2609  011526  000501                        BR      TRTEX           ;EXIT TEST
 2610  011530  000002                4$:     RTI                     ;RETURN FROM OVERFLOW TRAP
 2611
 2612  011532  010605                TRTC:   MOV     %6,%5           ;GET STACK POINTER
 2613  011534  162715  000002               SUB     #2,(5)          ;POINT PC TO TRAPPED INSTRUCTION
 2614  011540  013715  021416               MOV     TEMP+2,@(5)+    ;RESTORE ORIGINAL INSTRUCTION (COM TEMP)
 2615  011544  052715  000020               BIS     #20,(5)         ;SET 'T' BIT ON STACK
 2616  011550  012737  011566  000004       MOV     #TRTD,ERRVEC    ;CHANGE ERROR VECTOR
 2617  011556  012737  011702  000014       MOV     #TRTE,TRITVEC   ;LOAD 'T' BIT VECTOR
 2618  011564  000006                        RTT                     ;GO DO ORIGINAL INSTRUCTION AND TRAP
 2619                                                                 ;('T' BIT TRAP) WHEN FINISHED
 2620
 2621                                ;CHECK THE STACK
 2622  011566  022737  000031  000376 TRTD:  CMP     #31,376         ;IS 'N','C','T' BITS SET ON THE STACK
 2623  011574  001405                        BEQ     1$              ;(RESULT OF COM TEMP IN TRTA
 2624
 2625  011576  005212                        INC     (P2)            ;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 @R0" (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 @R0" (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 (TRITVEC+2) NOT SAVED
 2647  011652  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2648                                                                 ;"JMP @R0" (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 @R0" (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  000376        TRTE:   CLR     376             ;CLEAR SAVED 'T' BIT
 2663  011706  000002                        RTI                     ;RETURN FROM 'T' BIT TRAP
 2664
 2665  011710  022737  177777  021414 TRTF:  CMP     #-1,TEMP        ;WAS ORIGINAL INSTRUCTION EXECUTED?
 2666  011716  001405                        BEQ     TRTEX
 2667
 2668  011720  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2669  011722  012713  000214               MOV     #214,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2670  011726  000000                1$:     HLT                     ;TRACED INSTRUCTION (COM TEMP) NOT EXEC.
 2671  011730  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2672                                                                 ;"JMP @R0" (000110) TO GET A
 2673                                                                 ;SCOPE LOOP AND HIT CONTINUE
 2674  011732  012737  000016  000014 TRTEX: MOV     #TRTVEC+2,TRTVEC;RESTORE TRAP
 2675  011740  012737  000006  000004       MOV     #6,4            ;VECTORS
 2676  011746  010700                        SCOPE
 2677
 2678                                        ;*****************************************************************
 2679                                        ;*TEST 110 TEST YELLOW ZONE STACK VIOLATION, USING INDEX MODE (6) & R6. (R6.GT.4
 2680                                        ;*****************************************************************
 2681  011750  005237  000304        TST110: INC     @#STESTN
 2682  011754  012746  001000        INDEX:  MOV     #1000,%6        ;SET UP STACK POINTER
 2683  011760  012737  012010  000004       MOV     #IND1,4
 2684  011766  005037  000006               CLR     6
 2685  011772  005066  177340               CLR     -440(6)         ;FINAL ADDRESS IS 340
 2686
 2687  011776  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2688  012000  012713  000215               MOV     #215,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2689  012004  000000                        HLT                     ;ERROR! STACK OVERFLOW TRAP DID NOT OCCUR
 2690  012006  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2691                                                                 ;"JMP @R0" (000110) TO GET A
 2692                                                                 ;SCOPE LOOP AND HIT CONTINUE
 2693  012010  010704                IND1:   SCOPE
 2694
 2695                                ;TEST THAT THE TRAP SEQUENCE IS EXECUTED PROPERLY ON OVERFLOW.
 2696  012012  012706  001000               MOV     #1000,%6
 2697  012016  012737  012064  000004       MOV     #IND2,4
 2698  012024  005037  000776               CLR     776
 2699  012030  005037  000774               CLR     774
 2700  012034  012737  000001  000376       MOV     #1,376          ;PRE SET DEST ADDRESS
 2701  012042  012766  000000  177376       MOV     #0,-402(6)      ;FINAL ADDRESS IS 376
 2702  012050                        IND2B:
 2703
 2704  012050  005212                        INC     (P2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2705  012052  012713  000216               MOV     #216,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2706  012056  000000                IND2A:  HLT                     ;ERROR! FAILED TO TRAP
 2707  012060  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2708                                                                 ;"JMP @R0" (000110) TO GET A
 2709                                                                 ;SCOPE LOOP AND HIT CONTINUE
 2710  012062  000442                        BR      IND2X           ;GO TO SCOPE
 2711  012064  023727  000776  000004 IND2:   CMP     776,#4          ;STATUS SAVED ON THE STACK (Z BIT SET)
 2712  012072  001405                        BEQ     1$              ;[RESULT OF MOV #0,-402(6)]
 2713
 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
```

```
 2717  012104  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2718                                                                  ;"JMP @R0" (000110) TO GET A
 2719                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2720  012106  022737  012454  000774  1$:    CMP     #IND2R,774      ;RETURN PC SAVED?
 2721  012114  001405                        BEQ     2$
 2722
 2723  012116  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2724  012120  012713  000225                MOV     #225,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2725  012124  000000                        HLT                     ;ERROR! RETURN PC NOT SAVED
 2726  012126  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2727                                                                  ;"JMP @R0" (000110) TO GET A
 2728                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2729  012130  020627  000774          2$:    CMP     %6,#774         ;CORRECT STACK POINTER?
 2730  012134  001405                        BEQ     3$
 2731
 2732  012136  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2733  012140  012713  000221                MOV     #221,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2734  012144  000000                        HLT                     ;ERROR! INCORRECT STACK POINTER
 2735  012146  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2736                                                                  ;"JMP @R0" (000110) TO GET A
 2737                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2738  012150  005737  000376          3$:    TST     376
 2739  012154  001405                        BEQ     IND2X           ;WAS MOVE INST. EXECUTED
 2740
 2741  012156  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2742  012160  012713  000222                MOV     #222,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2743  012164  000000                        HLT                     ;ERROR! MOVE NOT EXECUTED
 2744  012166  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2745                                                                  ;"JMP @R0" (000110) TO GET A
 2746                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2747  012170  010700                  IND2X: SCOPE
 2748
 2749                                  ;*************************************************************
 2750                                  ;*TEST 111 TEST YELLOW ZONE STACK VIOLATION, USING INDEX MODE (340,LT,R6,LT,400
 2751                                  ;*************************************************************
 2752  012172  005237  000304  TST111: INC     @#$TESTN
 2753  012176  012706  000376          MOV     #376,%6 ;SET UP STACK IN "YELLOW" AREA
 2754  012202  012737  012236  000004  MOV     #IND3,4         ;SET UP ERROR RETURN
 2755  012210  005037  000374          CLR     374
 2756  012214  005237  000372          CLR     372
 2757  012220  005066  177742          CLR     -36(6)          ;FINAL ADDRESS IS 340 (YELLOW)
 2758  012224                  IND3B:
 2759
 2760  012224  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2761  012226  012713  000223                MOV     #223,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2762  012232  000000                  IND3A: HLT                     ;FAILED TO TRAP
 2763  012234  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2764                                                                  ;"JMP @R0" (000110) TO GET A
 2765                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2766  012236  022737  000004  000374  IND3:  CMP     #4,374          ;SAVED STATUS CORRECT?
 2767  012244  001405                        BEQ     1$
 2768
 2769  012246  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2770  012250  012713  000224                MOV     #224,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2771  012254  000000                        HLT                     ;SAVED STATUS WAS INCORRECT (Z SET)
 2772  012256  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
```



```
 2773                                                                  ;"JMP @R0" (000110) TO GET A
 2774                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2775  012260  022737  012224  000372  1$:    CMP     #IND3B,372      ;SAVED RETURN ADDRESS CORRECT?
 2776  012266  001405                        BEQ     2$
 2777
 2778  012270  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2779  012272  012713  000225                MOV     #225,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2780  012276  000000                        HLT                     ;ERROR! SAVED RETURN ADDRESS INCORRECT
 2781  012300  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2782                                                                  ;"JMP @R0" (000110) TO GET A
 2783                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2784  012302  022706  000372          2$:    CMP     #372,%6         ;STACK POINTER CORRECT
 2785  012306  001405                        BEQ     3$
 2786
 2787  012310  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2788  012312  012713  000226                MOV     #226,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2789  012316  000000                        HLT                     ;ERROR! STACK POINTER INCORRECT
 2790  012320  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2791                                                                  ;"JMP @R0" (000110) TO GET A
 2792                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2793  012322  005737  000340          3$:    TST     340             ;WAS CLEAR INSTRUCTION EXECUTED?
 2794  012326  001405                        BEQ     4$
 2795
 2796  012330  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2797  012332  012713  000227                MOV     #227,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2798  012336  000000                        HLT                     ;ERROR! CLEAR NOT EXECUTED
 2799  012340  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2800                                                                  ;"JMP @R0" (000110) TO GET A
 2801                                                                  ;SCOPE LOOP AND HIT CONTINUE
 2802  012342  010700                  4$:    SCOPE
 2803
 2804                                  ;*************************************************************
 2805                                  ;*TEST 112 TEST RED ZONE STACK VIOLATION & ABORT OF THE VIOLATING INSTRUCTION
 2806                                  ;*************************************************************
 2807  012344  005237  000304  TST112: INC     @#$TESTN
 2808  012350  012706  001000          MOV     #1000,%6
 2809  012354  012737  012436  000004  MOV     #IND4,4
 2810  012362  012737  177777  000344  MOV     #-1,344         ;PRE SET SAVED STATUS LOCATION
 2811  012370  012737  177777  000300  MOV     #-1,@
 2812  012376  012737  177777  000002  MOV     #-1,2
 2813  012404  005037  000776          CLR     776
 2814  012410  005037  177776          CLR     CC
 2815  012414  012737  177777  000336  MOV     #-1,336         ;PRE SET 336
 2816  012422  005066  177336          CLR     -442(6)         ;CLEAR "RED" LOCATION (336)
 2817  012426                  IND4B:
 2818
 2819  012426  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2820  012430  012713  000230                MOV     #230,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2821  012434  000000                  IND4A: HLT                     ;ERROR! FAILED TO TRAP
 2822  012436  005706                  IND4:  TST     %6              ;STACK POINTER CORRECT?
 2823  012440  001405                        BEQ     1$
 2824
 2825  012442  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2826  012444  012713  000231                MOV     #231,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2827  012450  000000                        HLT                     ;ERROR! INCORRECT STACK POINTER
 2828  012452  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
```

```
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
012454  022737  177777  000336  18:      CMP     #-1,336       ;WAS CLEAR INSTRUCTION ABORTED?
012462  001405                          BEQ     2$
                                        INC     (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
012464  005212
012466  012713  000232                  MOV     #232,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
012472  000000                          HLT                   ;ERROR! CLEAR INSTRUCTION NOT ABORTED
012474  000240                          LOOP                  ;REPLACE THIS INSTRUCTION BY A
                                                              ;"JMP @R0" (000110) TO GET A
                                                              ;SCOPE LOOP AND HIT CONTINUE
012476  022737  000004  000002  28:      CMP     #4,2          ;WAS PROPER STATUS SAVED? (Z SET)
012504  001405                          BEQ     3$            ;RESULT OF CLR -442(6)
                                        INC     (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
012506  005212
012510  012713  000233                  MOV     #233,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
012514  000000                          HLT                   ;ERROR! INCORRECT STATUS WAS SAVED
                                                              ;THOUGH THE INSTR. (CLR) WAS ABORTED
                                                              ;THE Z BIT WAS SET BECAUSE OF CLEAR
                                                              ;HENCE, ON THE STACK Z BIT WILL BE SAVED
012516  000240                          LOOP                  ;REPLACE THIS INSTRUCTION BY A
                                                              ;"JMP @R0" (000110) TO GET A
                                                              ;SCOPE LOOP AND HIT CONTINUE
012520  022737  012426  000000  38:      CMP     #IND4R,0      ;WAS RETURN PC SAVED?
012526  001405                          BEQ     4$
                                        INC     (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
012530  005212
012532  012713  000234                  MOV     #234,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
012536  000000                          HLT
012540  000240                          LOOP                  ;REPLACE THIS INSTRUCTION BY A
                                                              ;"JMP @R0" (000110) TO GET A
                                                              ;SCOPE LOOP AND HIT CONTINUE
012542  005737  000776          4$:     TST     776           ;WAS NOTHING PUT ON THE OLD STACK?
012546  001405                          BEQ     5$
                                        INC     (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
012550  005212
012552  012713  000235                  MOV     #235,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
012556  000000                          HLT                   ;ERROR! RED TRAP DID NOT STACK AT 2 & 0
012560  000240                          LOOP                  ;REPLACE THIS INSTRUCTION BY A
                                                              ;"JMP @R0" (000110) TO GET A
                                                              ;SCOPE LOOP AND HIT CONTINUE
012562  010700          5$:     SCOPE


                        ;*****************************************************************
                        ;*TEST 113 TEST THAT INDEX MODE USING R6 IN SOURCE DOES NOT TRAP
                        ;*****************************************************************
012564  005237  000304  TST113: INC     @#$TESTN
012570  012706  001000          MOV     #1000,%6      ;SET UP STACK
012574  012737  012626  000004          MOV     #IND5,4       ;SET UP ERROR TRAP
012602  012737  177777  000350          MOV     #-1,350
012610  016637  177350  021414          MOV     -430(6),TEMP
012616  000405                          BR      IND8
                                        INC     (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
012620  005212
012622  012713  000236                  MOV     #236,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
012626  000000          IND5:   HLT                   ;ERROR! TRAPPED
```



```
012630  000240                          LOOP                  ;REPLACE THIS INSTRUCTION BY A
                                                              ;"JMP @R0" (000110) TO GET A
                                                              ;SCOPE LOOP AND HIT CONTINUE
012632  010700          IND8:   SCOPE


                        ;*****************************************************************
                        ;*TEST 114 THIS TEST IS THE SAME AS ABOVE EXCEPT THAT LOCATION IS 'RED'
                        ;*****************************************************************
012634  005237  000304  TST114: INC     @#$TESTN
012640  012706  001000          MOV     #1000,%6
012644  012737  012670  000004          MOV     #IND6,4
012652  016637  177336  021414          MOV     -442(6),TEMP
012660  000405                          BR      IND9
                                        INC     (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
012662  005212
012664  012713  000237                  MOV     #237,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
012670  000000          IND6:   HLT                   ;ERROR! TRAPPED
012672  000240                          LOOP                  ;REPLACE THIS INSTRUCTION BY A
                                                              ;"JMP @R0" (000110) TO GET A
                                                              ;SCOPE LOOP AND HIT CONTINUE
012674  010700          IND9:   SCOPE


                        ;*****************************************************************
                        ;*TEST 115 TEST THAT TRAP DOES NOT OCCUR IF INDEX USING R6 IN DEST IS DEFFERED
                        ;*****************************************************************
012676  005237  000304  TST115: INC     @#$TESTN
012702  012706  001000          MOV     #1000,%6
012706  012737  012736  000004          MOV     #IND7,4
012714  012737  021414  000350          MOV     #TEMP,350
012722  005076  177350          CLR     @-430(6)              ;CLEAR TEMP
012726  000405                          BR      IND11
                                        INC     (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
012730  005212
012732  012713  000240                  MOV     #240,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
012736  000000          IND7:   HLT                   ;ERROR! TRAPPED
012740  000240                          LOOP                  ;REPLACE THIS INSTRUCTION BY A
                                                              ;"JMP @R0" (000110) TO GET A
                                                              ;SCOPE LOOP AND HIT CONTINUE
012742  010700          IND11:  SCOPE


                        ;*****************************************************************
                        ;*TEST 116 THIS TEST IS THE SAME AS ABOVE EXCEPT THE ADDRESS IS 'RED'
                        ;*****************************************************************
012744  005237  000304  TST116: INC     @#$TESTN
012750  012706  001000          MOV     #1000,%6
012754  012737  013006  000004          MOV     #IND10,4
012762  012737  021414  000336          MOV     #TEMP,336
012770  012776  177777  177336          MOV     #-1,@-442(6)          ;MOVE #-1 TO TEMP
012776  000405                          BR      IND12
                                        INC     (R2)          ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
013000  005212
013002  012713  000241                  MOV     #241,(R3)     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
013006  000000          IND10:  HLT                   ;ERROR TRAPPED
013010  000240                          LOOP                  ;REPLACE THIS INSTRUCTION BY A
                                                              ;"JMP @R0" (000110) TO GET A
                                                              ;SCOPE LOOP AND HIT CONTINUE
```

```
 2941  013012  010700                  IND12:  SCOPE
 2942
 2943                                          ;*************************************************************
 2944                                          ;*TEST 117 TEST RED ZONE VIOLATION USING EMT INSTRUCTION
 2945                                          ;*************************************************************
 2946  013014  005237  000304          TST117: INC     @#$TESTN
 2947  013020  012706  000340                  MOV     #340,%6         ;SET UP STACK POINTER IN 'RED' AREA
 2948  013024  012737  013112  000030          MOV     #RED1,EMTVEC
 2949  013032  005037  000032                  CLR     EMTVEC+2
 2950  013036  012737  013116  000004          MOV     #RED1A,4
 2951  013044  005037  000006                  CLR     6
 2952  013050  005037  000000                  CLR     @
 2953  013054  012737  136336  000336          MOV     #136336,336     ;PRESET 'RED' LOCATION
 2954  013062  012737  177777  000002          MOV     #-1,2           ;N BIT SET
 2955  013070  104000                  EMT
 2956  013072                          RED1C:
 2957
 2958  013072  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2959  013074  012713  000242                  MOV     #242,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2960  013100  000000                  RED1B:  HLT
 2961  013102  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2962                                          ;"JMP @R0" (000110) TO GET A
 2963                                          ;SCOPE LOOP AND HIT CONTINUE
 2964
 2965  013104  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2966  013106  012713  000243                  MOV     #243,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2967  013112  000000                  RED1:   HLT                     ;DID NOT ABORT EMT
 2968  013114  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 2969                                          ;"JMP @R0" (000110) TO GET A
 2970                                          ;SCOPE LOOP AND HIT CONTINUE
 2971  013116  022737  000000  000002  RED1A: CMP     @#0,2           ;WAS (NEW) STATUS SAVED?
 2972  013124  001405                  BEQ     1$
 2973
 2974  013126  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2975  013130  012713  000244                  MOV     #244,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2976  013134  000000                  HLT                             ;CORRECT (NEW) PSW WAS NOT SAVED
 2977                                          ;ON A RED ZONE TRAP (CAUSED BY EMT)
 2978                                          ;UPON EXECUTING EMT RED ZONE OCCURED
 2979                                          ;THE 11/60 SAVES THE "NEW" PSW WHICH
 2980                                          ;WOULD HAVE BEEN LOADED BY THE EMT.
 2981                                          ;11/40 SAVES THE PSW THAT WAS CURRENT AT THE
 2982                                          ;TIME EMT WAS EXECUTED.)
 2983  013136  000240                  LOOP                            ;REPLACE THIS INSTRUCTION BY A
 2984                                          ;"JMP @R0" (000110) TO GET A
 2985                                          ;SCOPE LOOP AND HIT CONTINUE
 2986  013140  022737  013072  000000  1$:     CMP     #RED1C,@        ;WAS RETURN PC SAVED
 2987  013146  001405                  BEQ     2$                      ;(EMTVEC)
 2988
 2989  013150  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 2990  013152  012713  000245                  MOV     #245,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 2991  013156  000000                  HLT
 2992  013160  000240                  LOOP                            ;REPLACE THIS INSTRUCTION BY A
 2993                                          ;"JMP @R0" (000110) TO GET A
 2994                                          ;SCOPE LOOP AND HIT CONTINUE
 2995  013162  005706                  2$:     TST     %6              ;STACK POINTER=0?
 2996  013164  001405                  BEQ     3$
```



```
 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                            ;REPLACE THIS INSTRUCTION BY A
 3002                                          ;"JMP @R0" (000110) TO GET A
 3003                                          ;SCOPE LOOP AND HIT CONTINUE
 3004  013200  022737  136336  000336  3$:     CMP     #136336,336     ;WAS 'RED' LOCATION UNDISTURBED?
 3005  013206  001405                  BEQ     4$
 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                            ;REPLACE THIS INSTRUCTION BY A
 3011                                          ;"JMP @R0" (000110) TO GET A
 3012                                          ;SCOPE LOOP AND HIT CONTINUE
 3013  013222  012737  000032  000030  4$:     MOV     #EMTVEC+2,EMTVEC
 3014  013230  010700                  SCOPE
 3015
 3016                                          ;*************************************************************
 3017                                          ;*TEST 120 TEST RED ZONE VIOLATION USING "TRAP" INSTRUCTION
 3018                                          ;*************************************************************
 3019  013232  005237  000304          TST120: INC     @#$TESTN
 3020  013236  012706  000200                  MOV     #200,%6         ;SET UP STACK IN 'RED' AREA
 3021  013242  012737  176176  000176          MOV     #176176,176     ;PRE SET 'RED' LOCATION
 3022  013250  012737  013326  000034          MOV     #RED2,TRAPVEC
 3023  013256  005037  000036                  CLR     TRAPVEC+2
 3024  013262  005037  000006                  CLR     6
 3025  013266  005037  000000                  CLR     @
 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                  RED2B:  HLT                     ;ERROR! FAILED TO TRAP
 3034  013316  000240                  LOOP                            ;REPLACE THIS INSTRUCTION BY A
 3035                                          ;"JMP @R0" (000110) TO GET A
 3036                                          ;SCOPE LOOP AND HIT CONTINUE
 3037
 3038  013320  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 3039  013322  012713  000251                  MOV     #251,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 3040  013326  000000                  RED2:   HLT                     ;DID NOT ABORT TRAP
 3041  013330  000240                  LOOP                            ;REPLACE THIS INSTRUCTION BY A
 3042                                          ;"JMP @R0" (000110) TO GET A
 3043                                          ;SCOPE LOOP AND HIT CONTINUE
 3044  013332  022737  000000  000002  RED2A:  CMP     @#0,2           ;WAS (NEW) STATUS SAVED?
 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                             ;CORRECT STATUS WORD WAS NOT SAVED
 3050                                          ;UPON A RED ZONE VIOLATION CAUSED BY
 3051                                          ;"TRAP".  11/60 SAVES THE NEW PSW
 3052                                          ;WHICH WOULD HAVE BEEN LOADED BY "TRAP"
```

```
3053
3054
3055  013352  000210                        LOOP                    ;11/40 SAVES THE "OLD" PSW THAT WAS CURRENT
3056                                                                  ;AT THE TIME "TRAP" WAS EXECUTED
3057                                                                  ;REPLACE THIS INSTRUCTION BY A
3058  013354  022737  013306  000000  1$:   CMP     #RED2C,0        ;"JMP @R0" (000110) TO GET A
3059  013362  001405                        BEQ     2$              ;SCOPE LOOP AND HIT CONTINUE
3060                                                                  ;(TRAPVEC)
3061  013364  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3062  013366  012713  000253                MOV     #253,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3063  013372  000000                        HLT
3064  013374  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
3065                                                                  ;"JMP @R0" (000110) TO GET A
3066                                                                  ;SCOPE LOOP AND HIT CONTINUE
3067  013376  005706                  2$:   TST     %6
3068  013400  001405                        BEQ     3$
3069
3070  013402  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3071  013404  012713  000254                MOV     #254,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3072  013410  000000                        HLT
3073  013412  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
3074                                                                  ;"JMP @R0" (000110) TO GET A
3075                                                                  ;SCOPE LOOP AND HIT CONTINUE
3076  013414  022737  176176  000176  3$:   CMP     #176176,176     ;WAS "RED" LOCATION LEFT UNDISTURBED?
3077  013422  001405                        BEQ     4$
3078
3079  013424  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3080  013426  012713  000255                MOV     #255,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3081  013432  000000                        HLT                     ;ERROR! "RED" LOCATION WAS CHANGED
3082  013434  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
3083                                                                  ;"JMP @R0" (000110) TO GET A
3084                                                                  ;SCOPE LOOP AND HIT CONTINUE
3085  013436  012737  000036  000034  4$:   MOV     #TRAPVEC+2,TRAPVEC
3086  013444  010700                        SCOPE
3087
3088                                       ;*********************************************************************
3089                                       ;*TEST 121 TEST RED ZONE VIOLATION USING "IOT" INSTRUCTION
3090                                       ;*********************************************************************
3091  013446  005237  000304          TST121: INC     @#$TESTN
3092  013452  012706  000100                  MOV     #100,%6
3093  013456  012737  013534  000020          MOV     #RED3,IOTVEC
3094  013464  005037  000022                  CLR     IOTVEC+2
3095  013470  005037  000006                  CLR     6
3096  013474  012737  013540  000004          MOV     #RED3A,4
3097  013502  005037  000000                  CLR     0
3098  013506  005037  000002                  CLR     2
3099  013512  000001                        IOT
3100  013514                           RED3C:
3101
3102  013514  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3103  013516  012713  000256                MOV     #256,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3104  013522  000000                  RED3B: HLT                     ;DID NOT OVERFLOW
3105  013524  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
3106                                                                  ;"JMP @R0" (000110) TO GET A
3107                                                                  ;SCOPE LOOP AND HIT CONTINUE
3108
```



```
3109  013526  005212                        INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3110  013530  012713  000257                MOV     #257,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3111  013534  000000                  RED3:  HLT *                   ;DID NOT ABORT IOT
3112  013536  000240                        LOOP                    ;REPLACE THIS INSTRUCTION BY A
3113                                                                  ;"JMP @R0" (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     1$
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 @R0" (000110) TO GET A
3128                                                                  ;SCOPE LOOP AND HIT CONTINUE
3129  013562  022737  013514  000000  1$:   CMP     #RED3C,0        ;WAS RETURN PC SAVED?
3130  013570  001405                        BEQ     2$              ;(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 @R0" (000110) TO GET A
3137                                                                  ;SCOPE LOOP AND HIT CONTINUE
3138  013604  005706                  2$:   TST     %6
3139  013606  001405                        BEQ     3$
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 @R0" (000110) TO GET A
3146                                                                  ;SCOPE LOOP AND HIT CONTINUE
3147  013622  012737  000022  000020  3$:   MOV     #IOTVEC+2,IOTVEC
3148  013630  010700                        SCOPE
3149
3150                                       ;*********************************************************************
3151                                       ;*TEST 122 TEST RED ZONE VIOLATION UPON TRACE TRAP
3152                                       ;*********************************************************************
3153  013632  005237  000304          TST122: INC     @#$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 ($FATAL) THE ERROR
3166  013704  000000                    HLT                     ;DID NOT OVERFLOW
3167  013706  000240            RED4B:  LOOP                    ;REPLACE THIS INSTRUCTION BY A
3168                                                            ;"JMP @R0" (000110) TO GET A
3169                                                            ;SCOPE LOOP AND HIT CONTINUE
3170
3171  013710  005212                    INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3172  013712  012713  000264            MOV     #264,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3173  013716  000000                    HLT                     ;DID NOT ABORT TRT
3174  013720  000240            RED4:   LOOP                    ;REPLACE THIS INSTRUCTION BY A
3175                                                            ;"JMP @R0" (000110) TO GET A
3176                                                            ;SCOPE LOOP AND HIT CONTINUE
3177  013722  022737  000000  000002   RED4A:  CMP     #0,2            ;WAS CORRECT PSW SAVED?
3178  013730  001405                    BEQ     1$
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 PSW TAH WOULD BE LOADED BY BPT
3185                                                            ;SHOULD BE SAVED.
3186  013742  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
3187                                                            ;"JMP @R0" (000110) TO GET A
3188                                                            ;SCOPE LOOP AND HIT CONTINUE
3189  013744  022737  013676  000004   1$:     CMP     #RED4C,@        ;WAS RETURN PC SAVED?
3190  013752  001405                    BEQ     2$              ;(TRTVEC)
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
3195  013764  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
3196                                                            ;"JMP @R0" (000110) TO GET A
3197                                                            ;SCOPE LOOP AND HIT CONTINUE
3198  013766  005726            2$:     TST     (6
3199  013770  001405                    BEQ     3$
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
3204  014002  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
3205                                                            ;"JMP @R0" (000110) TO GET A
3206                                                            ;SCOPE LOOP AND HIT CONTINUE
3207  014004  012737  000016  000014   3$:     MOV     #TPTVEC+2,TPTVEC
3208  014012  010700                    SCOPE
3209
3210                                    ;*****************************************************************
3211                                    ;*TEST 123 TEST TRANSITION FROM 'YELLOW' TO 'RED' AREAS
3212                                    ;*****************************************************************
3213  014014  005737  000304    TST123: INC     @#$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  000330    MOV     #RED5,@#EMTVEC  ;LOAD EMT VECTOR
3217  014032  005037  000032            CLR     @#EMTVEC+2      ;AND STATUS
3218  014036  012737  014132  000004    MOV     #RED5A,@#4      ;LOAD OVERFLOW VECTOR
3219  014044  012737  000001  000006    MOV     #1,@#6          ;AND STATUS
3220  014052  012737  000017  000002    MOV     #17,@#2
```



```
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                    RED5C:
3226
3227  014102  005212                    INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3228  014104  012713  000270            MOV     #270,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3229  014110  000000            RED5B:  HLT
3230  014112  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
3231                                                            ;"JMP @R0" (000110) TO GET A
3232                                                            ;SCOPE LOOP AND HIT CONTINUE
3233  014114  000240                    NOP
3234
3235  014116  005212                    INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3236  014120  012713  000271            MOV     #271,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3237  014124  000000            RED5:   HLT
3238  014126  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
3239                                                            ;"JMP @R0" (000110) TO GET A
3240                                                            ;SCOPE LOOP AND HIT CONTINUE
3241  014130  000240                    NOP
3242  014132  022737  000004  000342   RED5A:  CMP     #4,342          ;WAS STATUS SAVED?
3243  014140  001405                    BEQ     1$
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 @R0" (000110) TO GET A
3250                                                            ;SCOPE LOOP AND HIT CONTINUE
3251  014154  022737  014102  000340   1$:     CMP     #RED5C,340      ;WAS RETURN PC SAVED?
3252  014162  001405                    BEQ     2$
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 @R0" (000110) TO GET A
3259                                                            ;SCOPE LOOP AND HIT CONTINUE
3260  014176  022737  136336  000336   2$:     CMP     @#136336,@#336  ;WAS 'RED' LOCATION LEFT UNDISTURBED?
3261  014204  001405                    BEQ     3$
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 @R0" (000110) TO GET A
3268                                                            ;SCOPE LOOP AND HIT CONTINUE
3269  014220  022737  000001  000002   3$:     CMP     #1,2            ;WAS CORRECT PSW SAVED?
3270  014226  001405                    BEQ     4$
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
```

```
                                                        ;IS EXECUTED THE PENDING YELLOW ZONE OCCURS
                                                        ;(AND YELLOW ZONE PSW IS LOADED), BUT TRYING
                                                        ;TO PUSH THE STACK CAUSES RED ZONE AND
                                                        ;THE NEW PSW (YELLOW ZONE'S) SHOULD
                                                        ;BE SAVED ON THE STACK
014240  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014242  022737  014124  000000  4$: CMP     #RED5,0         ;WAS EMT'S VECTOR SAVED?
014250  001405                    BEQ     5$
                                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014252  005212                    INC     (R2)
014254  012713  000276            MOV     #276,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
014260  000000                    HLT                     ;ERROR! OVERFLOW'S VECTOR (4) NOT SAVED
014262  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014264  005706                5$: TST     %6
014266  001405                    BEQ     6$
014270  005212                    INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014272  012713  000277            MOV     #277,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
014276  000000                    HLT
014300  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014302                          6$:
014302  012737  000032  000030        MOV     @#EMTVEC+2,EMTVEC
014310  005037  000032            CLR     EMTVEC+2
014314  010700                    SCOPE
                                ;*****************************************************************
                                ;*TEST 124 TEST TRANSITION FROM 'YELLOW' TO 'RED' ZONES
                                ;*****************************************************************
014316  005237  000304      TST124: INC     @#$TESTN
                                ;THE TRANSITION OCCURS AFTER THE JSR HAS 'PUSHED' ITS OLD R5.
014322  012706  000342            MOV     #342,%6 ;SET UP STACK ON THE 'HAIRY EDGE'
014326  012737  014424  000004    MOV     #RED6A,4        ;LOAD OVERFLOW VECTOR
014334  012737  000357  000006    MOV     #357,6          ;AND OVERFLOW STATUS
014342  005037  000000            CLR     0
014346  005037  000002            CLR     2
014352  005037  000340            CLR     340
014356  012705  000007            MOV     #7,%5           ;PRE SET R5
014362  012737  000017  177776    MOV     #17,CC          ;PRE SET THE STATUS WORD
014370  004537  014416            JSR     5,RED6          ;SHOULD CAUSE OVERFLOW TRAP
014374                      RED6C:
                                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014374  005212                    INC     (R2)
014376  012713  000300            MOV     #300,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
014402  000000              RED6B: HLT                     ;ERROR! DID NOT OVERFLOW TRAP
014404  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014406  000240                    NOP
```



```
                                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014410  005212                    INC     (R2)
014412  012713  000301            MOV     #301,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
014416  000000              RED6:  HLT                     ;ERROR! DID NOT TRAP BEFORE JSR
014420  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014422  000240                    NOP
014424  022737  000357  177776  RED6A: CMP     #357,CC
014432  001405                    BEQ     6$
                                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014434  005212                    INC     (R2)
014436  012713  000302            MOV     #302,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
014442  000000                    HLT                     ;WRONG PSW ON RED ZONE TRAP
014444  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014446  022737  000007  000340  6$: CMP     #7,340          ;WAS R5 SAVED ON THE STACK
014454  001405                    BEQ     1$
                                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014456  005212                    INC     (R2)
014460  012713  000303            MOV     #303,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
014464  000000                    HLT                     ;ERROR! R5 NOT SAVED ON THE STACK
014466  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014470  022705  014374          1$:  CMP     #RED6C,%5       ;DOES R5 CONTAIN RETURN ADDRESS?
014474  001405                    BEQ     2$
                                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014476  005212                    INC     (R2)
014500  012713  000304            MOV     #304,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
014504  000000                    HLT                     ;ERROR! R5 DID NOT GET RETURN ADRS.
014506  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014510  022737  000357  000002  2$: CMP     #357,2          ;DID OVERFLOW SAVE THE STATUS?
014516  001405                    BEQ     3$
                                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014520  005212                    INC     (R2)
014522  012713  000305            MOV     #305,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
014526  000000                    HLT                     ;CORRECT PSW WAS NOT SAVED WHEN
                                                        ;A RED ZONE OCCURED WHILE TRYING
                                                        ;TO SERVICE YELLOW ZONE, AFTER THE JSR
                                                        ;IS EXECUTED THE PENDING YELLOW ZONE OCCURS
                                                        ;(AND YELLOW ZONE PSW IS LOADED), BUT TRYING
                                                        ;TO PUSH THE STACK CAUSES RED ZONE AND
                                                        ;THE NEW PSW (YELLOW ZONE'S) SHOULD
                                                        ;BE SAVED ON THE STACK
014530  000240                    LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
014532  022737  014416  000000  3$: CMP     #RED6,0         ;WAS RETURN ADDRESS SAVED ON THE STACK
014540  001405                    BEQ     4$
                                                        ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
014542  005212                    INC     (R2)
014544  012713  000306            MOV     #306,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
```

```
3389  014550  000000                          HLT                     ;OVERFLOW TRAP DID NOT SAVE RETURN PC
3390  014552  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3391                                                                  ;"JMP @R0" (000110) TO GET A
3392                                                                  ;SCOPE LOOP AND HIT CONTINUE
3393  014554  022737  136336  000336  4$:     CMP     #136336,336     ;WAS 'RED' LOCATION UNDISTURBED?
3394  014562  001405                          BEQ     5$
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 @R0" (000110) TO GET A
3401                                                                  ;SCOPE LOOP AND HIT CONTINUE
3402  014576  010700                  5$:     SCOPE
3403
3404                                          ;**********************************************************
3405                                          ;*TEST 125 TEST TRANSITION FROM 'YELLOW' TO 'RED' ZONE
3406                                          ;**********************************************************
3407  014600  005237  000304          TST125: INC     @#$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 R5
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                  RED7B:  HLT                     ;ERROR DID NOT OVERFLOW TRAP
3423  014662  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3424                                                                  ;"JMP @R0" (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                  RED7:   HLT                     ;ERROR! DID NOT TRAP BEFORE JSR
3430  014674  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3431                                                                  ;"JMP @R0" (000110) TO GET A
3432                                                                  ;SCOPE LOOP AND HIT CONTINUE
3433  014676  022737  000007  000342  RED7A:  CMP     #7,342          ;WAS R5 SAVED
3434  014704  001405                          BEQ     1$
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! R5 NOT SAVED ON THE STACK
3439  014716  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3440                                                                  ;"JMP @R0" (000110) TO GET A
3441                                                                  ;SCOPE LOOP AND HIT CONTINUE
3442  014720  022705  014652          1$:     CMP     #RED7C,%5       ;DOES R5 CONTAIN RETURN PC?
3443  014724  001405                          BEQ     2$
3444
```



```
3445  014726  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3446  014730  012713  000313                  MOV     #313,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3447  014734  000000                          HLT                     ;ERROR! JSR DID NOT LOAD R5
3448  014736  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3449                                                                  ;"JMP @R0" (000110) TO GET A
3450                                                                  ;SCOPE LOOP AND HIT CONTINUE
3451  014740  022737  000357  000340  2$:     CMP     #357,340        ;WAS STATUS SAVED?
3452  014746  001405                          BEQ     3$
3453
3454  014750  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3455  014752  012713  000314                  MOV     #314,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3456  014756  000000                          HLT                     ;ERROR! STATUS NOT SAVED
3457  014760  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3458                                                                  ;"JMP @R0" (000110) TO GET A
3459                                                                  ;SCOPE LOOP AND HIT CONTINUE
3460  014762  022737  000000  000002  3$:     CMP     #0,2            ;WAS STATUS SAVED?
3461  014770  001405                          BEQ     4$
3462
3463  014772  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3464  014774  012713  000315                  MOV     #315,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3465  015000  000000                          HLT                     ;CORRECT PSW WAS NOT SAVED WHEN
3466                                                                  ;A RED ZONE OCCURED WHILE TRYING
3467                                                                  ;TO SERVICE YELLOW ZONE. AFTER THE JSR
3468                                                                  ;IS EXECUTED THE PENDING YELLOW ZONE OCCURS
3469                                                                  ;(AND YELLOW ZONE PSW IS LOADED), BUT TRYING
3470                                                                  ;TO PUSH THE STACK CAUSES RED ZONE AND
3471                                                                  ;THE NEW PSW (YELLOW ZONE'S) SHOULD
3472                                                                  ;BE SAVED ON THE STACK
3473  015002  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3474                                                                  ;"JMP @R0" (000110) TO GET A
3475                                                                  ;SCOPE LOOP AND HIT CONTINUE
3476  015004  022737  014672  000000  4$:     CMP     #RED7,0         ;RETURN PC SAVED?
3477  015012  001405                          BEQ     5$
3478
3479  015014  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3480  015016  012713  000316                  MOV     #316,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3481  015022  000000                          HLT                     ;ERROR! RETURN PC NOT SAVED
3482  015024  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3483                                                                  ;"JMP @R0" (000110) TO GET A
3484                                                                  ;SCOPE LOOP AND HIT CONTINUE
3485  015026  022737  136336  000336  5$:     CMP     #136336,336     ;WAS 'RED' LOCATION UNDISTURBED?
3486  015034  001405                          BEQ     6$
3487
3488  015036  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3489  015040  012713  000317                  MOV     #317,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3490  015044  000000                          HLT                     ;ERROR! 'RED' LOCATION WAS CHANGED
3491  015046  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3492                                                                  ;"JMP @R0" (000110) TO GET A
3493                                                                  ;SCOPE LOOP AND HIT CONTINUE
3494  015050  010700                  6$:     SCOPE
3495
3496                                          ;**********************************************************
3497                                          ;*TEST 126 TEST THAT A TTY INTERRUPT CAUSES AN OVERFLOW TRAP
3498                                          ;**********************************************************
3499  015052  005237  000304          TST126: INC     @#$TESTN
3500  015056  012737  000340  177776          MOV     #340,STATUS     ;LOCK OUT INTERRUPT
```

```
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 @R0" (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  000003          TDEC8:  HLT                     ;TRAP FLAG OVERFLOW DID NOT OCCUR
3517  015140  000240                  LOOP                    ;REPLACE THIS INSTRUCTION BY A
3518                                                          ;"JMP @R0" (000110) TO GET A
3519                                                          ;SCOPE LOOP AND HIT CONTINUE
3520  015142  005037  177564  TDEC7:  CLR     TTCSR           ;CLEAR INTERRUPT ENABLE
3521  015146  010703                  SCOPE
```



```
3522
3523                                    ;*************************************************************
3524                                    ;*TEST 127 TEST THAT A TRAP OCCURS WHEN THE PC IS ODD
3525                                    ;*************************************************************
3526  015150  005237  000304  TST127: INC     @#$TESTN
3527  015154  012706  021774          MOV     #BUFF,%6                ;SET UP STACK POINTER
3528  015160  012737  015214  000004  MOV     #R7TR1,4                ;RETURN FROM TRAP
3529  015166  012707  000001          MOV     #1,%7                   ;PC EQUALS ONE
3530
3531  015172  005212                  INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3532  015174  012713  000322          MOV     #322,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3533  015200  000002                  HLT
3534  015202  000240                  LOOP                    ;REPLACE THIS INSTRUCTION BY A
3535                                                          ;"JMP @R0" (000110) TO GET A
3536                                                          ;SCOPE LOOP AND HIT CONTINUE
3537
3538  015204  005212                  INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3539  015206  012713  000323          MOV     #323,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3540  015212  000000                  HLT
3541  015214  022737  000003  021774  R7TR1:  CMP     #3,BUFF-4
3542  015222  001405                  BEQ     1$                      ;COPRECT 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 @R0" (000110) TO GET A
3549                                                          ;SCOPE LOOP AND HIT CONTINUE
3550
3551  015236  010700          1$:     SCOPE
3552
3553                                    ;*************************************************************
3554                                    ;*TEST 130 TEST THAT A TRAP OCCURS WHEN THE PC IS INCREMENTED TO AN ODD VALUE
3555                                    ;*************************************************************
3556  015240  005237  000304  TST130: INC     @#$TESTN
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 @R0" (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     1$                      ;COPRECT 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
```

```
3578  015322  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
3579                                                                    ;"JMP @R0" (000110) TO GET A
3580                                                                    ;SCOPE LOOP AND HIT CONTINUE
3581  015324  010700                    1$:   SCOPE
3582
3583                                          ;*****************************************************************
3584                                          ;*TEST 131 TEST THAT A DECREMENT OF PC TO AN ODD VALUE RESULTS IN A TRAP
3585                                          ;*****************************************************************
3586  015326  005237  000304           TST131: INC     @#$TESTN
3587  015332  012706  021774                   MOV     #BUFF,SP
3588  015336  012737  015360  000004           MOV     #R7TR3,4
3589  015344  005307                           DEC     PC                      ;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                             ;SHOULD TRAP
3595  015356  000240                           LOOP                    ;REPLACE THIS INSTRUCTION BY A
3596                                                                    ;"JMP @R0" (000110) TO GET A
3597                                                                    ;SCOPE LOOP AND HIT CONTINUE
3598  015360  022737  015347  021770   R7TR3:  CMP     #R7TR3A+1,BUFF-4        ;CHECK VALUE OF PC ON STACK
3599  015366  001405                           BEQ     1$
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 @R0" (000110) TO GET A
3606                                                                    ;SCOPE LOOP AND HIT CONTINUE
3607
3608  015402  010700                    1$:   SCOPE
3609
3610                                          ;*****************************************************************
3611                                          ;*TEST 132 TEST THAT "SEC, ROL PC" RESULTS IN TRAP
3612                                          ;*****************************************************************
3613  015404  005237  000304           TST132: INC     @#$TESTN
3614  015410  012706  021774                   MOV     #BUFF,SP
3615  015414  012737  015450  000004           MOV     #R7TR4,4
3616  015422  000761                           SEC                             ;CARRY EQUALS A 1
3617  015424  006107                           ROL     PC                      ;PC BECOMES ODD
3618  015426                           TR4A:
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 @R0" (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  015457  021770   R7TR4:  CMP     #TR4A+TP4A+3,BUFF-4     ;CHECK FOR VALUE ON STACK
3631  015456  001405                           BEQ     1$
3632
3633  015460  005212                           INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
```



```
3634  015462  012713  000334                   MOV     #334,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3635  015466  000000                           HLT                             ;WRONG VALUE ON STACK
3636  015470  000240                           LOOP                    ;REPLACE THIS INSTRUCTION BY A
3637                                                                    ;"JMP @R0" (000110) TO GET A
3638                                                                    ;SCOPE LOOP AND HIT CONTINUE
3639  015472  012737  000006  000004    1$:   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     @#$TESTN
3647  015506  012706  021774                   MOV     #BUFF,SP
3648  015512  012737  000340  177776           MOV     #340,STATUS             ;HIGHEST PRIOROTY LEVEL
3649  015520  012737  000100  177564           MOV     #100,TTCSR              ;INTERRUPT FOR TTY PUNCH/PRINTER
3650  015526  012737  015564  000034           MOV     #TR1,34                 ;TRAP" VECTOR
3651  015534  012737  015570  000064           MOV     #TR2,64                 ;TTY VECTOR
3652  015542  012737  000340  000036           MOV     #340,36                 ;IF TRAP TRAPS, MOVE 340 TO PRIORITY
3653  015550  005037  177776                   CLR     STATUS                  ;SHOULD TRAP AT IND OF CLR INST
3654  015554  104400                           TRAP                            ;TTY INTERRUPT SHOULD OVERPIDE TRAP
3655
3656  015556  005212                           INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3657  015560  012713  000335                   MOV     #335,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3658  015564  000000                   TR1:    HLT                             ;TRAP OCCUR FIRST
3659  015566  000240                           LOOP                    ;REPLACE THIS INSTRUCTION BY A
3660                                                                    ;"JMP @R0" (000110) TO GET A
3661                                                                    ;SCOPE LOOP AND HIT CONTINUE
3662  015570  005037  177564           TP2:    CLR     TTCSR
3663  015574  005037  000036                   CLR     36
3664  015600  010700                           SCOPE
3665
3666                                          ;*****************************************************************
3667                                          ;*TEST 134 TEST THAT A PENDING TTY INTERRUPT OCCURS BETWEEN "TRAP" & AN "IOT"
3668                                          ;*****************************************************************
3669  015602  005237  000304           TST134: INC     @#$TESTN
3670  015606  012706  021774                   MOV     #BUFF,SP
3671  015612  012737  000340  177776           MOV     #340,STATUS
3672  015620  012737  000100  177564           MOV     #100,TTCSR
3673  015626  012737  015664  000034           MOV     #TR3,34                 ;TRAP
3674  015634  005037  000036                   CLR     36
3675  015640  012737  015700  000064           MOV     #TR4,64                 ;TTY OUTPUT
3676  015646  012737  015674  000020           MOV     #TR5,20                 ;IOT
3677  015654  012737  000340  000022           MOV     #340,22                 ;IOT PRIORITY
3678  015662  104400                           TRAP                            ;THE ACT OF TRAPPING LOWER PRIORITY
3679  015664  000004                   TR3:    IOT                             ;INTERRUPT SHOULD OCCURE INPLACE OF IOT TRAP
3680
3681  015666  005212                           INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
3682  015670  012713  000336                   MOV     #336,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
3683  015674  000000                   TR5:    HLT                             ;NO INTERRUPT BETWEEN TRAPS
3684  015676  000240                           LOOP                    ;REPLACE THIS INSTRUCTION BY A
3685                                                                    ;"JMP @R0" (000110) TO GET A
3686                                                                    ;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
```

```
                                        ;*************************************************************
                                        ;*TEST 135 TEST THAT 'T' BIT TRAP OCCURS BEFORE AN INTERRUPT IS ACKNOWLEDGED
                                        ;*************************************************************
015712  005237  000304          TST135: INC     @#$TESTN
015716  012737  000340  177776          MOV     #340,CC         ;SET PRIORITY =7
015724  012737  000100  177564          MOV     #100,TTCSR      ;ENABLE INTERRUPT ON TTY PRINTER
015732  012737  015770  000014          MOV     #INT,TBITVEC    ;LOAD 'T' BIT VECTOR
015740  005037  000016                  CLR     TBITVEC+2
015744  012737  016002  000064          MOV     #INT2,64        ;LOAD TTY INT. VECTOR
015752  012706  021774                  MOV     #BUFF,%6        ;SET STACK POINTER
015756  012746  000020                  MOV     #20,-(6)                ;PUSH 'T'BIT ON THE STACK
015762  012746  015770                  MOV     #.+6,-(6)               ;PUSH PC ON THE STACK
015766  000002                          RTI                     ;SET 'T' BIT
015770                          INT:
                                        ;
015770  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
015772  012713  000337                  MOV     #337,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
015776  000000                          HLT                     ;ERROR! TTY INTERRUPTED
016000  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
016002  005037  177564          INT2:   CLR     TTCSR           ;DISABLE TTY INTERRUPT
016006  022737  000020  021772          CMP     #20,BUFF-2
016014  001405                          BEQ     1$
                                        ;
016016  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016020  012713  000340                  MOV     #340,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016024  000000                          HLT
016026  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
016030  022737  015770  021770  1$:     CMP     #INT,BUFF-4
016036  001405                          BEQ     2$
                                        ;
016040  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016042  012713  000341                  MOV     #341,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016046  000000                          HLT
016050  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
016052  022737  000000  021766  2$:     CMP     #0,BUFF-6
016060  001405                          BEQ     3$
                                        ;
016062  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016064  012713  000342                  MOV     #342,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016070  000000                          HLT
016072  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
016074  022737  015770  021764  3$:     CMP     #INT,BUFF-10
016102  001405                          BEQ     INT1
                                        ;
016104  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016106  012713  000343                  MOV     #343,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016112  000000                          HLT
016114  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
```



```
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
016116  010700                  INT1:   SCOPE
                                        ;*************************************************************
                                        ;*TEST 136 TEST THAT "RESET" GOES TO OUTSIDE WORLD
                                        ;*************************************************************
016120  005237  000304          TST136: INC     @#$TESTN
016124  012706  021774                  MOV     #BUFF,%6
016130  012737  000340  177776          MOV     #340,CC         ;LOCK OUT INTERRUPTS
016136  012737  016206  000064          MOV     #TR6,64                 ;LOAD TELEPRINTER VECTOR
016144  012737  000100  177564          MOV     #100,TTCSR              ;SET INTERRUPT ENABLE
016152  000005                          RESET                   ;SHOULD CLEAR INTERPUPT ENABLE
016154  032737  000100  177564          BIT     #100,TTCSR              ;TEST FOR CLEAR
016162  001405                          BEQ     1$
                                        |
016164  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016166  012713  000344                  MOV     #344,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016172  000000                          HLT                     ;RESET FAILED TO CLEAR TTCSR
016174  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
016176  000406                  1$:     BR      TR6X            ;GO TO SCOPE
                                        ;
016200  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016202  012713  000345                  MOV     #345,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016206  000000                  TR6:    HLT                     ;ERROR! TELETYPE INTERRUPTED WHEN
                                        ;                       ;PROCESSER WAS AT LEVEL 7
016210  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                        ;                       ;"JMP @R0" (000110) TO GET A
                                        ;                       ;SCOPE LOOP AND HIT CONTINUE
016212  022626                          CMP     (6)+,(6)+
016214  005037  177564          TR6X:   CLR     TTCSR
016220  010700                          SCOPE
                                        ;*************************************************************
                                        ;*TEST 137 TEST THAT RESET DOES NOT HANG THE SYSTEM
                                        ;*************************************************************
016222  005237  000304          TST137: INC     @#$TESTN
016226  012706  021774                  MOV     #BUFF,%6                ;SET STACK
016232  005037  177776                  CLR     STATUS                  ;ALLOW INTERRUPT
016236  012737  016252  000064          MOV     #RESET1,64              ;TTY INTERRUPT VECTOR
016244  052737  000100  177564          BIS     #100,TTCSR              ;SET INTERRUPT ENABLE
016252  000005                  RESET1: RESET                   ;IF THIS HANGS CHECK SACK
016254  012737  000066  000064          MOV     #66,64                  ;FOR FALSE INTERRUPT
016262  010700                          SCOPE
                                        ;*************************************************************
                                        ;*TEST 140 TEST RESET WITH TRACE ON
                                        ;*************************************************************
016264  005237  000304          TST140: INC     @#$TESTN
016270  012706  021774                  MOV     #BUFF,%6                ;SET STACK
016274  012737  016336  000014          MOV     #RESET2,TRITVEC ;SET UP TRACE VECTOR
016302  005037  000016                  CLR     TBITVEC+2
016306  012746  000020                  MOV     #20,-(6)                ;PUSH 'T'BIT ON THE STACK
016312  012746  016320                  MOV     #.+6,-(6)               ;PUSH PC ON THE STACK
016316  000002                          RTI                     ;SET 'T' BIT
```

```
 3802  016320  000005                          RESET                           ;SHOULD HAVE NO EFFECT
 3803  016322  000005                          RESET                           ;NO EFFECT
 3804
 3805  016324  005212                          INC     (P2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 3806  016326  012713  000346                  MOV     #346,(P3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 3807  016332  000000                          HLT                             ;TRACE TRAP FAILED
 3808  016334  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 3809                                                                  ;"JMP @R0" (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     @#$TESTN
 3819  016354  000005                          RESET
 3820  016356  012706  021774                  MOV     #BUFF,%6                ;SET UP STACK
 3821  016362  012737  016406  000064          MOV     #TTY3,64                ;INTERRUPT VECTOR
 3822  016370  005037  177776                  CLR     STATUS                  ;DROP PROCESSOR PRIORITY
 3823  016374  012737  000357  000066          MOV     #357,66                 ;HIGH PRIORITY ON INTERRUPT
 3824  016402  005137  177564                  COM     TTCSR                   ;SHOULD SET INTERRUPT ENABLE & INTERRUPT
 3825
 3826  016406  013727  177776          TTY3:   MOV     STATUS,(PC)+            ;SAVE PROCESSOR STATUS
 3827  016412  000000                          .WORD   0
 3828  016414  022737  000357  016412          CMP     #357,.-2
 3829  016422  001405                          BEQ     1$
 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 @R0" (000110) TO GET A
 3836                                                                  ;SCOPE LOOP AND HIT CONTINUE
 3837  016436  000005                  1$:     RESET                           ;CLR INTERRUPT ENABLE
 3838  016440  010700                          SCOPE
 3839  016442  012706  021774                  MOV     #BUFF,%6                ;STACK SET UP
 3840  016446  012737  016472  000064          MOV     #TTY4,64                ;INTERRUPT VECTOR
 3841  016454  005037  000066                  CLR     66                      ;CLR NEW STATUS
 3842  016460  012737  000157  177776          MOV     #157,STATUS             ;PROCESSOR STATUS
 3843  016466  005137  177564                  COM     TTCSR                   ;SET INTERRUPT ENABLE
 3844  016472  013727  177776          TTY4:   MOV     STATUS,(PC)+            ;SAVE NEW STATUS
 3845  016476  000000                          .WORD   0
 3846  016500  005737  016476                  TST     .-2
 3847  016504  001405                          BEQ     1$
 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 @R0" (000110) TO GET A
 3854                                                                  ;SCOPE LOOP AND HIT CONTINUE
 3855  016520  005037  177564          1$:     CLR     TTCSR
 3856  016524  010700                          SCOPE
 3857                                          ;*****************************************************************
```



```
 3858
 3859                                          ;*TEST 142 CONTIGUOUS MEMORY ADDRESS TEST
 3860  016526  005237  000304          TST142: INC     @#$TESTN
                                                ;*****************************************************************
 3861
 3862  016532  012706  021774          ;THIS TEST CHECKS THAT ALL MEMORY (UP TO 28K) IS CONTIGUOUS
 3863  016536  012737  016552  000004          MOV     #BUFF,%6                ;SET STACK POINTER
 3864  016544  005000                          MOV     #MEMEND,@#4             ;SET TIME OUT TRAP VECTOR
 3865  016546  005720                          CLR     %0                      ;SET STARTING ADDRESS FOR TEST
 3866  016550  003776                          TST     (0)+                    ;BEGIN
 3867  016552  022626                          BR      .-2                     ;LOOP UNTIL TIMEOUT OCCURS
 3868  016554  022700  160002          MEMEND: CMP     (6)+,(6)+               ;RESTORE STACK POINTER
 3869  016560  001406                          CMP     #160002,%0              ;AT END OF 28K MEMORY??
 3870  016562  005720                          BEQ     MEMEX                   ;YES - EXIT TEST
 3871                                          TST     (0)+                    ;SHOULD TIME OUT HERE UP TO 28K
 3872  016564  005212                          INC     (R2)            ;SET MESSAGE TYPE ($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 @R0" (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
```

```
                                        ;SPECIAL CASE OF ODD:,EVEN ,BYTE AND REGISTER 6

                                                ;*****************************************************************
                                                ;*TEST 143 TEST AUTO INCREMENT & DECREMENT OF R6 - WORD & BYTE
                                                ;*****************************************************************
016612  005237  000304          TST143: INC     @#$TESTN
016616  012706  021774          AIDR6:  MOV     #BUFF,%6        ;SET UP THE STACK
016622  112627                          MOVB    (6)+,(PC)+      ;SIX SHOULD INCREMENT BY TWO
016624  000000                          .WORD   0
016626  020627  021776                  CMP     %6,#BUFF+2
016632  001405                          BEQ     1$

016634  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016636  012713  000352                  MOV     #352,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016642  000000                          HLT                     ;R6 DID NOT AUTO INCREMENT BY TWO
016644  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
016646  010700                  1$:     SCOPE

016650  012706  001000                  MOV     #1000,%6
016654  114627                          MOVB    -(6),(PC)+      ;SHOULD DECREMENT BY TWO
016656  000000                          .WORD   0
016660  020627  000776                  CMP     %6,#776
016664  001405                          BEQ     2$

016666  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016670  012713  000353                  MOV     #353,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016674  000000                          HLT                     ;R6 DID NOT AUTO DECRENT BY 2
016676  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
016700  010700                  2$:     SCOPE

016702  012706  021774                  MOV     #BUFF,%6
016706  112626                          MOVB    (6)+,(6)+       ;DOUBLE AUTO INCREMENT OF R6
016710  020627  022000                  CMP     %6,#BUFF+4
016714  001405                          BEQ     3$

016716  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016720  012713  000354                  MOV     #354,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
016724  000000                          HLT                     ;WRONG AUTO INCREMENT OF R6
016726  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
016730  010700                  3$:     SCOPE

016732  012706  021774                  MOV     #BUFF,%6
016736  005004                          CLR     %4
016740  122624                          CMPB    (6)+,(4)+       ;TEST INCREMENT OF R6
016742  020627  021776                  CMP     %6,#BUFF+2
016746  001405                          BEQ     4$

016750  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
016752  012713  000355                  MOV     #355,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
```



```
016756  000000                          HLT                     ;WRONG INCREMENT OF R6
016760  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
016762  010700                  4$:     SCOPE

016764  012706  021774                  MOV     #BUFF,%6
016770  005004                          CLR     %4
016772  122426                          CMPB    (4)+,(6)+       ;TEST INCREMENT OF R6
016774  020627  021776                  CMP     %6,#BUFF+2
017000  001405                          BEQ     5$

017002  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
017004  012713  000356                  MOV     #356,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
017010  000000                          HLT                     ;WRONG INCREMENT OF R6
017012  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
017014  010700                  5$:     SCOPE

017016  012706  021774                  MOV     #BUFF,%6
017022  005004                          CLR     %4
017024  122624                          CMPB    (6)+,(4)+       ;TEST INCREMENT OF R4
017026  020427  000001                  CMP     %4,#1
017032  001405                          BEQ     6$

017034  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
017036  012713  000357                  MOV     #357,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
017042  000000                          HLT                     ;WRONG INCREMENT OF R4
017044  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
017046  010700                  6$:     SCOPE

017050  012706  021774                  MOV     #BUFF,%6
017054  005004                          CLR     %4
017056  122446                          CMPB    (4)+,-(6)       ;TEST DECREMENT OF R6
017060  020627  021772                  CMP     %6,#BUFF-2
017064  001405                          BEQ     7$

017066  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
017070  012713  000360                  MOV     #360,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
017074  000000                          HLT                     ;WRONG INCREMENT OF R6
017076  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
017100  010700                  7$:     SCOPE

017102  012706  021774                  MOV     #BUFF,%6
017106  005004                          CLR     %4
017110  122426                          CMPB    (4)+,(6)+       ;TEST INCREMENT OF R4
017112  020427  000001                  CMP     %4,#1
017116  001405                          BEQ     8$

017120  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
017122  012713  000361                  MOV     #361,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
017126  000000                          HLT                     ;WRONG INCREMENT OF R4
```

```
 3993  017130  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 3994                                                                  ;"JMP @R0" (000110) TO GET A
 3995                                                                  ;SCOPE LOOP AND HIT CONTINUE
 3996  017132  010700                    5$:   SCOPE
 3997
 3998  017134  012706  001000                   MOV     #1000,R6
 3999  017140  174627                           CMPB    -(6),(PC)+      ;TEST DECREMENT OF R6
 4000  017142  022706  000776                   CMP     #776,R6
 4001  017146  001405                           BEQ     9$
 4002
 4003  017150  005212                           INC     (R2)
 4004  017152  012713  000362                   MOV     #362,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4005  017156  000000                           HLT                     ;WRONG DECREMENT OF R6
 4006  017160  000240                           LOOP                    ;REPLACE THIS INSTRUCTION BY A
 4007                                                                  ;"JMP @R0" (000110) TO GET A
 4008                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4009  017162  010700                    9$:   SCOPE
 4010
 4011  017164  010700                          SCOPE
 4012  017166  012706  001774                   MOV     #BUFF,R6        ;5 JAN 71
 4013  017172  012737  177777  001772           MOV     #-1,BUFF-2      ;INITIAL POSITION OF R6
 4014  017200  012737  025252  001000           MOV     #25252,K1       ;HIGH BYTE WILL BE DATA
 4015  017206  114637  001001                   MOVB    -(6),K1+1       ;MOV TO ODD ADDRESS
 4016  017212  022737  177652  001000           CMP     #177652,K1      ;DID MOVB WORK CORRECTLY
 4017  017220  001405                           BEQ     10$
 4018
 4019  017222  005212                           INC     (R2)
 4020  017224  012713  000363                   MOV     #363,(R3)       ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4021  017230  000000                           HLT                     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4022  017232  000240                           LOOP                    ;MOVB -(6), ODD: FAILED
 4023                                                                  ;REPLACE THIS INSTRUCTION BY A
 4024                                                                  ;"JMP @R0" (000110) TO GET A
 4025  017234  010700                   10$:   SCOPE                   ;SCOPE LOOP AND HIT CONTINUE
 4026
 4027  017236  012706  001774                   MOV     #BUFF,R6        ;INITIAL POSITION OF R6
 4028  017242  012737  177777  001772           MOV     #-1,BUFF-2      ;HIGH BYTE SOURCE DATA
 4029  017250  012737  125252  001000           MOV     #125252,K1      ;INITIAL SET UP OF DESTINATION
 4030  017256  114637  001000                   MOVB    -(6),K1         ;HIGH BYTE OF BUFF-2,TO LOW BYTE OF K1
 4031  017262  022737  125377  001000           CMP     #125377,K1      ;TEST RESULTS
 4032  017270  001405                           BEQ     11$
 4033
 4034  017272  005212                           INC     (R2)
 4035  017274  012713  000364                   MOV     #364,(R3)       ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4036  017300  000000                           HLT                     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4037  017302  000240                           LOOP                    ;MOVB -(6), EVEN, FAILED
 4038                                                                  ;REPLACE THIS INSTRUCTION BY A
 4039                                                                  ;"JMP @R0" (000110) TO GET A
 4040  017304  010700                   11$:   SCOPE                   ;SCOPE LOOP AND HIT CONTINUE
```



```
 4041
 4042                                          ;*********************************************************************
 4043                                          ;*TEST 144 TEST TRANSFER OF .BYTE USING R6
 4044  017306  005237  000304          TST144: ;*********************************************************************
 4045  017312  012737  123456  001010  BXFR:   INC     @#$TESTN
 4046  017320  012737  050505  001000          MOV     #123456,K5
 4047  017326  012705  001000                  MOV     #050505,K1
 4048  017332  012706  001010                  MOV     #K1,R5          ;R5=(050505)K1
 4049  017336  112625                          MOV     #K5,R6          ;R6(123456)K5
 4050  017340  022737  050456  001000          MOVB    (6)+,(5)+       ;LOW .BYTE OF R6 TO R5
 4051  017346  001405                          CMP     #050456,K1
 4052                                          BEQ     1$
 4053  017350  005212
 4054  017352  012713  000365                  INC     (R2)
 4055  017356  000000                          MOV     #365,(R3)       ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4056  017360  000240                          HLT                     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4057                                          LOOP                    ;FALSE TRANSFER OF .BYTE
 4058                                                                  ;REPLACE THIS INSTRUCTION BY A
 4059  017362  010700                    1$:   SCOPE                   ;"JMP @R0" (000110) TO GET A
 4060                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4061  017364  012737  123456  001010          MOV     #123456,K5
 4062  017372  012737  050505  001000          MOV     #050505,K1
 4063  017400  012705  001000                  MOV     #K1,R5          ;R5(050505)K1
 4064  017404  012706  001012                  MOV     #K6,R6          ;R6(123456)K5
 4065  017410  114625                          MOVB    -(6),(5)+       ;LOW .BYTE OF R6 TO R5 (DECREMENT)
 4066  017412  023727  001000  050456          CMP     K1,#050456
 4067  017420  001405                          BEQ     2$
 4068
 4069  017422  005212
 4070  017424  012713  000366                  INC     (R2)
 4071  017430  000000                          MOV     #366,(R3)       ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4072  017432  000240                          HLT                     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4073                                          LOOP                    ;FALSE R6 .BYTE TRANSFER
 4074                                                                  ;REPLACE THIS INSTRUCTION BY A
 4075  017434  010700                    2$:   SCOPE                   ;"JMP @R0" (000110) TO GET A
 4076                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4077  017436  012737  123456  001000          MOV     #123456,K1
 4078  017444  012737  050505  001010          MOV     #050505,K5
 4079  017452  012705  001000                  MOV     #K1,R5          ;(123456)
 4080  017456  012706  001010                  MOV     #K5,R6          ;(050505)
 4081  017462  112526                          MOVB    (5)+,(6)+       ;LOW OF R5 TO LOW OF R6
 4082  017464  022737  050456  001010          CMP     #050456,K5
 4083  017472  001405                          BEQ     3$
 4084
 4085  017474  005212                          INC     (R2)
 4086  017476  012713  000367                  MOV     #367,(R3)       ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4087  017502  000000                          HLT                     ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4088  017504  000240                          LOOP                    ;FALSE R6 .BYTE TRANSFER
 4089                                                                  ;REPLACE THIS INSTRUCTION BY A
 4090                                                                  ;"JMP @R0" (000110) TO GET A
 4091  017506  010700                    3$:   SCOPE                   ;SCOPE LOOP AND HIT CONTINUE
 4092
 4093  017510  012737  123456  001000          MOV     #123456,K1
 4094  017516  012737  050505  001010          MOV     #050505,K5
 4095  017524  012705  001001                  MOV     #K1+1,R5        ;123456
 4096  017530  012706  001010                  MOV     #K5,R6          ;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 @R0" (000110) TO GET A
 4106                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4107  017560  010700                  4$:     SCOPE
 4108
 4109  017562  012737  123456  001000          MOV     #123456,K1
 4110  017570  012737  050505  001010          MOV     #050505,K5
 4111  017576  012705  001001                  MOV     #K1+1,R5        ;R5-123456--ODD ADDRESS
 4112  017602  012706  001010                  MOV     #K5,R6          ;R6-050505--.EVEN ADDRESS
 4113  017606  112625                          MOVB    (6)+,(5)+       ;LOW OF R6 TO HIGH OF R5
 4114  017610  022737  042456  001000          CMP     #042456,K1
 4115  017616  001405                          BEQ     5$
 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 @R0" (000110) TO GET A
 4122                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4123  017632  010700                  5$:     SCOPE
 4124
 4125                                          ;*************************************************************
 4126                                          ;*TEST 145 TEST .BYTE OPERATION WITH SEQUENTIAL ODD;.EVEN ADDRESS
 4127                                          ;*************************************************************
 4128  017634  005237  000304          TST145: INC     @#$TESTN
 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
    6                                                                 ;"JMP @R " (       ) 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 @R0" (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
```



```
 4153
 4154  017720  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4155  017722  012713  000374                  MOV     #374,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4156  017726  000000                          HLT                     ;ODD TO .EVEN FAILED
 4157  017730  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 4158                                                                  ;"JMP @R0" (000110) TO GET A
 4159                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4160  017732  010700                  3$:     SCOPE
 4161
 4162  017734  123737  001016  001012          CMPB    K10,K6
 4163  017742  001405                          BEQ     4$
 4164
 4165  017744  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4166  017746  012713  000375                  MOV     #375,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4167  017752  000000                          HLT                     ;.EVEN TO EVEN FAILED
 4168  017754  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 4169                                                                  ;"JMP @R0" (000110) TO GET A
 4170                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4171  017756  010700                  4$:     SCOPE
 4172
 4173  017760  123737  001015  001017          CMPB    K7+1,K10+1
 4174  017766  001405                          BEQ     5$
 4175
 4176  017770  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4177  017772  012713  000376                  MOV     #376,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4178  017776  000000                          HLT                     ;ODD TO ODD FAILED
 4179  020000  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 4180                                                                  ;"JMP @R0" (000110) TO GET A
 4181                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4182  020002  010700                  5$:     SCOPE
 4183
 4184  020004  123737  001016  001017          CMPB    K10,K10+1
 4185  020012  001005                          BNE     6$
 4186
 4187  020014  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4188  020016  012713  000377                  MOV     #377,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4189  020022  000000                          HLT                     ;LOW TO HIGH IN SAME .WORD FAILED
 4190  020024  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 4191                                                                  ;"JMP @R0" (000110) TO GET A
 4192                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4193  020026  010700                  6$:     SCOPE
 4194
 4195  020030  123737  001017  001016          CMPB    K10+1,K10
 4196  020036  001005                          BNE     7$
 4197
 4198  020040  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
 4199  020042  012713  000400                  MOV     #400,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
 4200  020046  000000                          HLT                     ;HIGH TO LOW IN SAME .WORD FAILED
 4201  020050  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 4202                                                                  ;"JMP @R0" (000110) TO GET A
 4203                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4204  020052  010700                  7$:     SCOPE
 4205
 4206  020054  123737  001016  001015          CMPB    K10,K7+1
 4207  020062  001005                          BNE     8$
 4208
```

```
 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 @R0" (000110) TO GET A
 4214                                                                ;SCOPE LOOP AND HIT CONTINUE
 4215  020076  010700                  8$:   SCOPE
 4216
```



```
 4217
 4218
 4219                                        ;*****************************************************************
 4220                                        ;*TEST 146 TEST THAT MOVB %R MOVES ONLY THE LSH OF THE REGISTER
 4221  020100  005237  000304         TST146: INC     @#$TESTN        ;*****************************************************************
 4222  020104  012737  177777  001020 RLOBYT: MOV     #-1,K11
 4223  020112  012700  000125                MOV     #125,%0         ;LOAD R0
 4224  020116  110037  001020                MOVB    %0,K11
 4225  020122  023727  001020  177525        CMP     K11,#177525     ;WAS ONLY LSH MOVED?
 4226  020130  001405                        BEQ     1$
 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 @R0" (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 @R0" (000110) TO GET A
 4247                                                                ;SCOPE LOOP AND HIT CONTINUE
 4248  020202  010700                  2$:   SCOPE
 4249
 4250  020204  012706  001020                MOV     #K11,%6
 4251  020210  010637  001020                MOV     %6,K11
Z 4252  020214  110626                        MOVB    %6,(6)+
 4253  020216  023727  001020  001022        CMP     K11,#K11+2
 4254  020224  001405                        BEQ     3$
 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 @R0" (000110) TO GET A
 4261                                                                ;SCOPE LOOP AND HIT CONTINUE
 4262  020240  010700                  3$:   SCOPE
 4263
 4264  020242  012706  001020                MOV     #K11,%6
Z 4265  020246  010626                        MOV     %6,(6)+
 4266  020250  023727  001020  001022        CMP     K11,#K11+2
 4267  020256  001405                        BEQ     4$
 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
```

```
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020272  010700                  4$:     SCOPE
020274  000277                          SCC             ;SET STATUS
020276  005037  177776                  CLR     STATUS  ;CLEAR STATUS
020302  103005                          BCC     5$
020304  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020306  012713  000406                  MOV     #406,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020312  000000                          HLT             ;C NOT CLEAR
020314  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020316  102405                  5$:     BVC     6$
020320  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020322  012713  000407                  MOV     #407,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020326  000000                          HLT             ;V NOT CLEAR
020330  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020332  001005                  6$:     BNE     7$
020334  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020336  012713  000410                  MOV     #410,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020342  000000                          HLT             ;Z NOT CLEAR
020344  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020346  100005                  7$:     BPL     8$
020350  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020352  012713  000411                  MOV     #411,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020356  000000                          HLT             ;N NOT CLEAR
020360  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020362  010700                  8$:     SCOPE
020364  000257                          CCC             ;CLEAR CONDITION CODES
020366  052737  000017  177776          BIS     #17,STATUS ;SET STATUS TO ONES
020374  103405                          BCS     9$
020376  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020400  012713  000412                  MOV     #412,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020404  000000                          HLT             ;C NOT SET
020406  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020410  102405                  9$:     BVS     10$
020412  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020414  012713  000413                  MOV     #413,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020420  000000                          HLT             ;V NOT SET
```



```
020422  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020424  001405                  10$:    BEQ     11$
020426  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020430  012713  000414                  MOV     #414,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020434  000000                          HLT             ;Z NOT SET
020436  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020440  100405                  11$:    BMI     12$
020442  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020444  012713  000415                  MOV     #415,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020450  000000                          HLT             ;N NOT SET
020452  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020454  010700                  12$:    SCOPE

                                                        ;TEST THAT (%6) +4 TO %6
020456  012706  021774                  MOV     #BUFF,%6 ;SET UP BUFFER POINTER
020462  005037  021776                  CLR     BUFF+2  ;SET UP NEW STATUS
020466  012737  020476  021774          MOV     #.+10,BUFF ;SET UP, RETURN
020474  000006                          RTT             ;RETURN NEXT INSTRUCTION
020476  020627  022000                  CMP     %6,#BUFF+4 ;%6 SHOULD BE PLUS 4
020502  001405                          BEQ     13$
020504  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020506  012713  000416                  MOV     #416,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020512  000000                          HLT             ;%6 NOT INCREMENTED
020514  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020516  010700                  13$:    SCOPE
020520  012737  000357  021776          MOV     #357,BUFF+2 ;RETURN STATUS ALL ONES
020526  012706  021774                  MOV     #BUFF,%6 ;SET UP BUFFER POINTER
020532  012737  020544  021774          MOV     #.+12,BUFF ;SET UP RETURN POINTER
020540  000257                          CCC             ;CLEAR CONDITION CODES
020542  000006                          RTT             ;RETURN

020544  001405                          BEQ     14$

020546  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020550  012713  000417                  MOV     #417,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
020554  000000                          HLT             ;Z NOT SET
020556  000240                          LOOP            ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
020560  102405                  14$:    BVS     15$

020562  005212                          INC     (R2)    ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
020564  012713  000420                  MOV     #420,(R3) ;MOVE TO MAIL BOX ($FATAL) THE ERROR
```

```
 4385  020574  000000                          HLT                     ;V NOT SET
 4386  020572  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 4387                                                                  ;"JMP @R0" (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 @R0" (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 @R0" (000110) TO GET A
 4404                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4405
 4406  020624  010700                  17$:    SCOPE
 4407  020626  005037  021776                  CLR     BUFF+2          ;ZERO TO STATUS
 4408  020632  012706  021774                  MOV     #BUFF,SP
 4409  020636  012737  020650  021774          MOV     #.+12,BUFF
 4410  020644  000777                          SCC                     ;SET CONDITION CODES
 4411  020646  000006                          RTT                     ;RETURN
 4412  020650  001005                          BNE     18$
 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 @R0" (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 @R0" (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 @R0" (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
```



```
 4441  020726  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
 4442                                                                  ;"JMP @R0" (000110) TO GET A
 4443                                                                  ;SCOPE LOOP AND HIT CONTINUE
 4444  020730  010700                  21$:    SCOPE
 4445  020732                          RLOBY1:
```

```
                                        ;*****************************************************************
                                        ;*TEST 147 TEST THAT ALL RESERVED INSTRUCTIONS TRAP
                                        ;*****************************************************************
020732  005237  000304          TST147: INC     @#$TESTN
020736  012700  137777                  MOV     #137777,R0              ;DISABLE MCS IF PRESENT
020742  076600                          MED                             ;BY DOING "INIT".  XFC SHOULD
020744  000352                          352                             ;TRAP AA A RESERVED INSTR.
020746  012700  021344                  MOV     #TABLE,TAB              ;TABLE POINTER
020752  012002                  GIN1:   MOV     (TAB)+,FIRST            ;FIRST OR CURRENT INSTRUCTION
020754  012001                          MOV     (TAB)+,LAST             ;LAST INSTRUCTION OF GROUP
020756  020237  021400                  CMP     FIRST,FINISH            ;TESTED ALL
020762  001453                          BEQ     RET3A           ;GO TO MULT TRAPS TEST IF FIN.
020764  010237  021402          GIN2:   MOV     FIRST,INST              ;SET UP INST
020770  012737  021012  000010          MOV     #RET,RESVEC     ;LOAD RESERVED INST. TRAP VECTOR
020776  012706  021774                  MOV     #BUFF,%6                ;SET UP STACK POINTER
021002  005037  177776                  CLR     @#177776                ;CLEAR PRIORITY
021006  000137  021402                  JMP     INST                    ;EXECUTE RESERVED INSTRUCTION
                                                                        ;TRAP SHOULD OCCUR

                                        ;TRAPPING SHOULD SEND YOU HERE
021012  020627  021770          RET:    CMP     %6,#BUFF-4              ;TEST DECREMENT OF %6
021016  001404                          BEQ     RET1
021020  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021022  012713  000427                  MOV     #427,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021026  000000                          HLT                             ;WRONG DECREMENT
021030  023727  021770  021404  RET1:   CMP     BUFF-4,#INST+2          ;LOC OF INST UNINCREMENTED
021036  001405                          BEQ     RET2
021040  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021042  012713  000430                  MOV     #430,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021046  000000                          HLT                             ;INST INC ON TRAP
021050  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
021052  005737  021772          RET2:   TST     BUFF-2
021056  001405                          BEQ     RET3

021060  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021062  012713  000431                  MOV     #431,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021066  000000                          HLT                             ;CONDITION CODES SET ON TRAP
021070  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
021072  005237  021402          RET3:   INC     INST
021076  005202                          INC     FIRST
021100  023701  021402                  CMP     INST,LAST
021104  001722                          BEQ     GIN1                    ;SET UP NEW GROUP
021106  000137  020764                  JMP     GIN2                    ;FINISH OLD GROUP
                                                                        ;END OF INSTRUCTION GROUP
021112  010700                  RET3A:  SCOPE

                                        ;*****************************************************************
                                        ;*TEST 150 CHECK MULTIPLE TRAP CONDITIONS
                                        ;*****************************************************************
```



```
021114  005237  000304          TST150: INC     @#$TESTN
                                        ;THE FOLLOWING TESTS ARE WRITTEN TO CHECK MULTIPLE TRAP CONDITIONS
                                        ;TO TEST THAT TRAPS ARE PROCESSED PROPERLY.
021120  012706  021774          SINS1:  MOV     #BUFF,%6                ;SET UP STACK POINTER
021124  012737  021176  000014          MOV     #SINSD,TBITVEC  ;LOAD 'T' BIT VECTOR
021132  012737  021222  000004          MOV     #SINSE,4        ;LOAD ERROR VECTOR
021140  012737  021205  000030          MOV     #SINSF+1,EMTVEC ;LOAD EMT VECTOR (ODD)
021146  005037  000032                  CLR     EMTVEC+2
021152  005000                          CLR     %0                      ;PRE SET R0
021154  012746  000020                  MOV     #20,-(6)                ;SET UP
021160  012746  021166                  MOV     #SINSDD,-(6)            ;TO SET 'T' BIT
021164  000006                          RTT                             ;SET 'T' BIT
021166  104000          SINSDD: EMT                             ;DO EMT PICK UP ODD PC

021170  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021172  012713  000432                  MOV     #432,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021176  000000          SINSD:  HLT                             ;ERROR! 'T' TRAP OCCURRED BEFORE ERROR
021200  000240                  LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                        ;"JMP @R0" (000110) TO GET A
                                                        ;SCOPE LOOP AND HIT CONTINUE
021202  000455                          BR      SINS1           ;GO TO SCOPE
021204  000240          SINSF:  NOP                             ;THE EMT TRIES TO COME HERE

021206  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021210  012713  000433                  MOV     #433,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021214  000000                          HLT                             ;ERROR! NO ERROR TRAP ON ODD EMT VECTOR
021216  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
021220  000446                          BR      SINSI           ;GO TO SCOPE
021222  012737  021246  021764  SINSE:  MOV     #SINSG,BUFF-10  ;CORRECT BAD EMT VECTOR ON THE STACK
021230  000006                          RTT                             ;GO TO SINSG (SERVICE EMT)

021232  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021234  012713  000434                  MOV     #434,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021240  000000                          HLT                             ;DIDN'T GO
021242  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
021244  000434                          BR      SINSI           ;GO TO SCOPE
021246  012737  021276  021770  SINSG:  MOV     #SINSH,BUFF-4   ;RESTORE RETURN FROM EMT (ON STACK)
021254  012737  021316  000014          MOV     #SINSJ,TBITVEC  ;CHANGE 'T' BIT VECTOR
021262  000006                          RTT                             ;GO TO SINSH

021264  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021266  012713  000435                  MOV     #435,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021272  000000                          HLT
021274  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
021276  012700  000001          SINSH:  MOV     #1,%0           ;LOAD INDICATOR

021302  005212                          INC     (R2)            ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021304  012713  000436                  MOV     #436,(R3)       ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021310  000000                          HLT                             ;ERROR! SHOULD HAVE 'T' TRAPPED
021312  000240                          LOOP                    ;REPLACE THIS INSTRUCTION BY A
```

```
                                                                ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
021314  000410                        BR      SINSI           ;GO TO SCOPE
021316  022700  000001        SINSJ:  CMP     #1,R0           ;WAS INST. AT SINSH EXECUTED?
021322  001405                        BEQ     SINSI
                                                                ;SET MESSAGE TYPE ($MSGTY) TO FATAL ERROR
021324  005212                        INC     (R2)            ;MOVE TO MAIL BOX ($FATAL) THE ERROR
021326  012713  000437                MOV     #437,(R3)       ;ERROR! INST AT SINSH WAS NOT EXECUTED
021332  000000                        HLT                     ;REPLACE THIS INSTRUCTION BY A
021334  000240                        LOOP                    ;"JMP @R0" (000110) TO GET A
                                                                ;SCOPE LOOP AND HIT CONTINUE
021336  010700                SINSI:  SCOPE
021340  000137  021420                JMP     ENDOT           ;END OF PROGRAM, GOT TO
                                                                ;END OF TEST LINKAGE
                                                                ;XFC= 076XXX EXCEPT 076XX
021344  000007                TABLE:  7
                                                                ;MED=076600
021346  000077                        77
021350  000210                        210
021352  000227                        227
021354  007000                        7000
021356  007777                        7777
021360  075040                        75040
021362  076577                        76577
021364  076601                        76601
021366  076677                        76677
021370  106400                        106400
021372  106477                        106477
021374  106700                        106700
021376  107777                        107777
021400  021400                FINISH: .                               ;END FLAG
021402  000000                INST:   HALT                            ;WILL CONTAIN RESERVED INST
021404  000000                        HALT                            ;SHOULD TRAP TO LOC 10
021406  000000                        HALT                            ;LOC 10 SHOULD SEND YOU TO
021410  000000                        HALT                            ;RET
021412  000000                        HALT
021414  000000                TEMP:   0
        021420                        .=.+2

                                ;END OF PASS LINKAGE

021420  005237  000306        ENDOT:  INC     @#$PASS
021424  105237  021546                INCB    PASSPT          ;SHOULD PRINT THIS PASS?
021430  001027                        BNE     ACT             ;NO
021432  132737  000040  000321        BITB    #40,$ENVM       ;WILL APT ALLOW PRINTING?
021440  001023                        BNE     ACT             ;NO
021442  023727  000042  021520        CMP     @#42,@#ENDAD
021450  001417                        BEQ     ACT
021452  012700  021550                MOV     #MSG,R0         ;GET MSG ADDR.
```



```
021456  105737  177564        WAIT:   TSTB    @#TPS           ;TTY READY
021462  100375                        BPL     WAIT            ;NO WAIT
021464  112037  177566                MOVB    (R0)+,@#TPB     ;PRINT CHARACTER
021470  001372                        BNE     WAIT            ;NEXT IF NOT DONE.
021472  105737  177564        WAIT1:  TSTB    @#TPS
021476  100375                        BPL     WAIT1
021500  000005                        RESET
021502  012737  177761  021546        MOV     #177761,PASSPT  ;DO IT ABOUT 15 DECIMAL TIMES
021510  013700  000042        ACT:    MOV     @#42,R0         ;CHECK ACT
021514  001405                        BEQ     GOAGIN          ;KEEP GOING
021516  000005                        RESET
021520  004710                $ENDAD: JSR     PC,(R0)         ;ACT HOOKS
021522  000240                        NOP
021524  000240                        NOP
021526  000240                        NOP
021530  012737  000012  000010  GOAGIN: MOV     #12,10
021536  005037  000012                CLR     12
021542  000137  001110                JMP     BEGIN1          ;DO NEXT PASS
021546  177777                PASSPT: -1
021550  005015  047105  020104  MSG:   .ASCIZ  <15><12>/END OF  PASS/
021556  043117  020040  040520
021564  051523     000
        021570                        .EVEN

                                ;POWER FAIL, POWER UP ROUTINE

021570  012737  021600  000024  PWRDWN: MOV     #PWRUP,24
021576  000000                        HALT

021600  012737  021570  000024  PWRUP:  MOV     #PWRDWN,24
021606  012706  021774                MOV     #BUFF,SP
021612  132737  000040  000321        BITB    #40,$ENVM       ;WILL APT ALLOW PRINTING?
021620  001013                        BNE     PFRES           ;NO
021622  012700  021654                MOV     #MSGPWF,R0      ;GET MSG ADDR.
021626  105737  177564        PWAIT:  TSTB    @#TPS           ;TTY READY
021632  100375                        BPL     PWAIT           ;NO WAIT
021634  112037  177566                MOVB    (R0)+,@#TPB     ;PRINT CHARACTER
021640  001372                        BNE     PWAIT           ;NEXT IF NOT DONE.
021642  105737  177564        PWAIT1: TSTB    @#TPS
021646  100375                        BPL     PWAIT1
021650  000137  001102        PFRES:  JMP     BEGIN
021654  005015  047520  042527  MSGPWF: .ASCIZ  <15><12>.POWER FAILED!.
021662  020122  040506  046111
021670  042105  000041
        021774                .=.+100
021774  000000                BUFF:   0
        000001                        .END
```

```
ABASE = 000000            318
ACDW1 = 000000            318
ACDW2 = 000000            318
ACPUOP= 000000            318      333
ACT     021510            346#     4610     4612     4622#
ADDW0 = 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      374
ADEVM = 000000            318
AENV  = 000000            318      329
AENVM = 000000            318      330
AFATAL= 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
AMSGLG= 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            343      397#     4654
BEGIN1  001113            401#     4631
BFLI  = 000240            286#
BGN2    001202            418#     419      421
BGN2A   001216            423#     424
BGN3    001224            413      416      425#
```

```
BPT    = 000033          273#  2568
BPTVEC= 000014           279*  1017*  1018*  1034*  1053*  1072*  1088*  1109*  1110*  1127*  1128*  1144*
BUFF    021774           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    923    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   3527   3541   3557   3572   3587   3598   3614   3630   3647
                        3670   3699   3712   3721   3730   3739   3754   3785   3796   3820   3839   3862   3888
                        3891   3916   3918   3929   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#
BXFR    017312           4045#
BYTOP   017640           4129#
CC     = 177776          294#   499*   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   1264*  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#
EMTVEC= 000030           277#   862*   863*   879*   898*   917*   933*   954*   955*   971*   972*   1008*  1009*
                        2331*  2332*  2948*  2949*  3013*  3216*  3217*  3305*  3306*  4508*  4509*
ENDOT   021420           4572   4606#
EPRVEC= 000001           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*  1509*  1510*  1526*  1527*  1694*  1695*  1843*  1844*  1862*  1879*  1896*  1910*
                        1930*  2339*  2340*  2565*  2566*  2616*
FINISH  021400           4457   4594#
FOVER   010503           2276   2299#
FOV1    010503           2295   2306#
GIN1    020752           4455#  4494
GIN2    020764           4459#  4495
GOAGIN  021530           4623   4629#
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   2149   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#   2970
ILLB   = 000100          293#   2981
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*
ITRAPS= 000004           274#
K1      001400           372#   4014*  4015*  4016   4029*  4030*  4031   4046*  4047   4050   4062*  4063   4066
                        4077*  4079   4093*  4095   4109*  4111   4114
K10     001016           379#   4151   4162   4173   4184   4195   4206
K11     001020           380#   4222*  4224*  4225   4236   4237*  4239   4250   4251*  4253   4264   4266
K12     001022           381#
K2      001002           373#
K3      001004           374#
K4      001006           375#
K5      001010           376#   4045*  4048   4061*  4078*  4080   4082   4094*  4096   4098   4110*  4112
K6      001012           377#   4064   4162
K7      001014           378#   4129   4140   4151   4173   4206
LOOP   = 000240          287#    440    460    479    499    517    539    559    579    598    617    636    653
                         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
```

```
                    2790    2799    2828    2837    2649    285R    2867    2885    2902    2920    2938    2961    2968
                    2983    2992    3001    3010    3034    3041    3055    3064    3073    3082    3105    3112    3126
                    3135    3144    3167    3174    3186    3195    3204    3230    3238    3248    3257    3266    3282
                    3291    3300    3329    3337    3347    3356    3365    3381    3390    3399    3423    3430    3439
                    3448    3457    3473    3482    3491    3510    3517    3534    3547    3565    3578    3595    3604
                    3623    3636    3659    3684    3708    3718    3727    3736    3745    3765    3774    3808    3834
                    3852    3875    3897    3911    3924    3938    3952    3966    3979    3993    4006    4022    4037
                    4056    4072    408R    4104    4120    4135    4146    4157    4168    4179    4190    4201    4212
                    4231    4245    4259    4272    42R4    4292    4300    4348    4321    4329    4337    4345    4362
                    4378    4386    4394    4402    4417    4425    4433    4441    4479    4488    4519    4528    4538
                    4549    4557    4567
MED    = 076600      298#    4452
MEMEND  016552      3863    3867#
MFMFX   016576      3869    3878#
MSG     021550      4613    4633#
MSGNAM  001026       386#     417
MSGPWF  021654      4647    4655#
NAMFLG  001024       383#     412     414*
NOP    = 000240       287     288#
OAER0   005736      1547    1558#
OAER1   005774      1561    1571#
OAER2   006042      1578    1589#
OAER3   006104      1596    1606#
OAER4   006136      1609    1619#
OAER5   006204      1626    1637#
OAER6   006324      1644    1672#
OAER7   006366      1683    1692#
OAER8   006330      1668    1676#
00      010616      2342#
00A     010626      2347#
001     010620      2343#    2402
01      010632      2331    2351#    2422
01A     010642      2357#
011     010634      2353#    2442
02      010646      2333    2361#    2462
02A     010656      2367#
021     010650      2363#    2482
03      010662      2335    2371#    2502
03A     010672      2377#
031     010664      2373#    2522
04      010676      2337    2391#    2542
05      011324      2339    2554#
06      011330      2553    2555#
07      011322      2543    2553#
PASSPT  021546       397*    4607*    4621*    4632#
PFRES   021650      4646    4654#
PWAIT   021626      4648#    4649    4651
PWAIT1  021642      4652#    4653
PWRDWN  021570       402    4640#    4643
PWRUP   021600      4640    4643#
RA      003502       993    1004#
RA1     002344       703     714#
RB      003466       992*    995#    1004*    1005
RB1     002330       702*    705#     714*     715
RC      003462       994#    1006
RC1     002324       704#     716
```



```
RED1    013112      2948    2967#
RED1A   013116      2950    2971#
RED1B   013100      2960#
RED1C   013072      2956#    2986
RED2    013326      3022    3040#
RED2A   013332      3026    3044#
RED2B   013314      3033#
RED2C   013306      3029#    3058
RED3    013534      3093    3111#
RED3A   013540      3096    3115#
RED3B   013522      3104#
RED3C   013514      3100#    3129
RED4    013716      3155    3173#
RED4A   013722      3158    3177#
RED4B   013704      3166#
RED4C   013676      3162#    3189
RED5    014124      3216    3237#    3285
RED5A   014132      3218    3242#
RED5B   014110      3229#
RED5C   014102      3225#    3251
RED6    014416      3323    3336#    3384
RED6A   014424      3316    3341#
RED6B   014402      3328#
RED6C   014374      3324#    3359
RED7    014672      3417    3429#    3476
RED7A   014676      3411    3433#
RED7B   014660      3422#
RED7C   014652      3418#    3442
RESET1  016252      3787    3789#
RESET2  016336      3797    3811#
RESINS= 000007       290#     435     451     471     490     507     530     549
RESVEC= 000010       291#     433*    434*    450*    470*    488*    505*    527*    528*    546*    547*    563*    564*
                    4460*
RET     021012      4460    4467#
RETA    001264       433     443#
RETAT   006454      1705    1718#
RETA1   001716       572     582#
RETA2   002432       727     737#
RETA3   003064       862     872#
RETA4   003570      1017    1027#
RETA5   004226      1154    1164#
RETB    001306       450     452#
RETBT   006524      1735    1739#
RETB1   001740       589     591#
RETB2   002454       744     746#
RETB3   003106       879     881#
RETB4   003612      1034    1036#
RETB5   004250      1171    1173#
RETC    001350       470     472#
RETCT   006610      1754    1767#
RETCT1  006724      1795    1810#
RETCT2  006712      1802#    1810
RETCT3  006576      1758#    1767
RETC1   002002       608     610#
RETC2   002516       763     765#
RETC3   003150       898     900#
```

```
RETC4   #03654          1053    1055#
RETC5   #04312          1190    1192#
RETD    #01420           488     491#
RETD1   #02052           626     629#
RFTD2   #02566           782     785#
RETD3   #03220           917     920#
RETD4   #03724          1072    1075#
RETD5   #04362          1209    1212#
RETE    #01464           505     508#
RETE1   #02116           642     645#
RETE2   #02632           798     801#
RETE3   #03264           933     936#
RETE4   #03770          1088    1091#
RFTF5   #04426          1225    1228#
RFTF    #01544           527     531#
RETF1   #02176           663     667#
RFTF2   #02712           819     823#
RETF3   #03344           954     958#
RETF4   #04050          1109    1113#
RETF5   #04506          1246    1250#
RFTG    #01620           546     550#
RETG1   #02252           680     684#
RETG2   #02766           837     841#
RETG3   #03414           971     975#
RETG4   #04124          1127    1131#
RFTG5   #04562          1264    1268#
RETH5   #04652          1287    1297#
RFTJ    #04670          1301    1303#
RFTK    #04732          1320    1322#
RETL    #05002          1339    1342#
RFTM    #05046          1355    1358#
RETN    #05126          1376    1380#
RFTO    #05202          1394    1398#
RFTP    #05274          1418    1428#
RETQ    #05320          1436    1438#
RETR    #05364          1455    1457#
RETS    #05436          1474    1477#
RETT    #05504          1489    1492#
RETU    #05566          1509    1513#
RETV    #05640          1526    1529#
RET1    #21030          4468    4473#
RET2    #21052          4474    4482#
RFT3    #21072          4483    4491#
RET3A   #21112          4458    4497#
RLOBYT  #20104          4222#
RLOBY1  #20732          4445#
R7TP1   #15214          3528    3541#
R7TP2   #15302          3550    3572#
R7TP2A  #15260          3560#   3572
R7TP3   #15360          3588    3598#
R7TP3A  #15346          3590#   3598
R7TR4   #15450          3615    3630#
SCOPE = #10700           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
```



```
                        1410    1428    1448    1467    1486    1502    1523    1540    1558    1571    1589    1606    1619
                        1637    1676    1696    1729    1748    1787    1836    1855    1872    1888    1907    1921    1956
                        1978    1999    2019    2040    2061    2082    2092    2113    2150    2179    2208    2237    2266
                        2306    2555    2676    2693    2747    2802    2870    2888    2905    2923    2941    3014    3086
                        3148    3208    3307    3402    3494    3521    3551    3581    3608    3641    3664    3689    3748
                        3779    3791    3813    3838    3856    3880    3900    3914    3927    3941    3955    3969    3982
                        3996    4009    4011    4025    4040    4059    4075    4091    4107    4123    4138    4149    4160
                        4171    4182    4193    4204    4215    4234    4248    4262    4275    4311    4348    4366    4406
                        4444    4497    4570
SINSD   #21176           4506    4518#
SINSDD  #21166           4512    4514#
SINSE   #21222           4507    4532#
SINSF   #21204           4508    4523#
SINSG   #21246           4532    4542#
SINSH   #21276           4542    4552#
SINSI   #21336           4522    4531    4541    4560    4562    4570#
SINSJ   #21316           4543    4561#
SINS1   #21120           4505#
STATUS= 177776            289#   3500#   3505*   3648*   3653*   3671*   3786*   3811*   3822*   3826    3842*   3844    4278*
                         4314*
                         2547#
STKERR  #11316           4454    4576#
TABLE   #21344            276#   2617*   3696*   3697*   3797*   3798*   4506*   4543*
TBITVE= 000014           1843    1850#
TDEC1   #07034           1846    1855#
TDEC1A  #07040           1862    1872#
TDEC2   #07076           1879    1888#
TDEC2A  #07134           1896    1902#
TDEC3   #07172           1910    1916#
TDEC4   #07224           1930    1941#
TDEC5   #07304           1936#
TDEC5A  #07276           1940    1955    1956#
TDEC5B  #07336           1898    1907#
TDEC6   #07176           1912    1921#
TDEC6A  #07230           3502    3520#
TDEC7   #15142           3503    3516#
TDEC8   #15136           1646    1647    1648*   1649*   1684*   2561*   2567*   2570*   2614    2665    2879*   2896*   2913
TFMP    #21414           2931    4600#
                          285#    420*   4616*   4650*
TPB   = 177566            284#    418     423     4614    4618    4648    4652
TPS   = 177564            281#    572*    573*    589*    600*    626*    642*    663*    664*    680*    681*    719*    720*
TRAPVE= 000034           2333*   2334*   3022*   3023*   3085*
                          283#
TRCSR = 177560            272#
TRT   = 000003           2567    2568*   2570*   2581
TRTA    #11410           2571#   2631
TRTAA   #11414           2565    2572#
TRTB    #11416           2563    2600    2612#
TRTC    #11532           2616    2622#
TRTD    #11566           2617    2650    2662#
TRTE    #11702           2590    2609    2640    2659    2666    2674#
TRTEX   #11732           2571    2665#
TRTF    #11710            278#   1705*   1706*   1735*   1754*   1795*   1834*   1835*   2337*   2338*   2563*   2564*   2674*
TRTVEC= 000014           3155*   3156*   3207*
TR1     #15564           3650    3658#
TR2     #15570           3651    3662#
```

```
TR3     015664          3673    3679#
TP4     015700          3675    3687#
TP4A    015426          3618#   3630
TR5     015674          3676    3683#
TR6     016206          3756    3772#
TP6X    016214          3768    3778#
TST1    001226           431#
TST10   001762           646#
TST100  010030          2126#
TST101  010106          2155#
TST102  010164          2184#
TST103  010242          2213#
TST104  010320          2242#
TST105  010376          2273#
TST106  010506          2311#
TST107  011332          2566#
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          2916#
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          3596#
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          3896#
TST144  017306          4044#
TST145  017634          4128#
TST146  020100          4221#
TST147  020732          4450#
TST15   002434           742#
TST150  021114          4502#
TST16   002476           761#
TST17   002542           780#
```



```
TST2    001266           448#
TST20   002660           817#
TST21   003026           860#
TST22   003066           877#
TST23   003130           896#
TST24   003174           915#
TST25   003312           952#
TST26   003442           991#
TST27   003532          1015#
TST3    001334           468#
TST30   003572          1032#
TST31   003634          1051#
TST32   003700          1070#
TST33   004016          1107#
TST34   004170          1152#
TST35   004230          1169#
TST36   004272          1188#
TST37   004336          1207#
TST4    001374           486#
TST40   004454          1244#
TST41   004614          1285#
TST42   004712          1318#
TST43   004756          1337#
TST44   005074          1374#
TST45   005234          1416#
TST46   005276          1434#
TST47   005342          1453#
TST5    001512           525#
TST50   005410          1472#
TST51   005532          1507#
TST52   005672          1545#
TST53   005776          1576#
TST54   006044          1594#
TST55   006140          1624#
TST56   006206          1642#
TST57   006332          1681#
TST6    001660           578#
TST60   006402          1702#
TST61   006474          1733#
TST62   006546          1752#
TST63   006656          1793#
TST64   007000          1841#
TST65   007042          1860#
TST66   007100          1877#
TST67   007136          1893#
TST7    001720           587#
TST70   007232          1926#
TST71   007340          1962#
TST72   007412          1983#
TST73   007464          2003#
TST74   007536          2024#
TST75   007610          2045#
TST76   007662          2066#
TST77   007756          2097#
TTCSR = 177564           282#   3504#   3520#   3649#   3662#   3672#   3687#   3695#   3711#   3757#   3759    3778#   3788#
                        3824#   3843#   3855#
```

```
TTY3    016406          3821    3826#
TTY4    016472          3840    3844#
VDEC1   007410          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          2215    2254#
VDEC2   007404          1964    1974#
VDEC3   007462          1986    1999#
VDEC4   007456          1985    1995#
VDEC5   007534          2006    2019#
VDEC6   007530          2005    2015#
VDEC7   007606          2027    2040#
VDEC8   007602          2026    2036#
VDEC9   007666          2048    2061#
WAIT    021456          4614#   4615    4617
WAIT1   021472          4618#   4619
$APTHD  000330           351     357#
$CPUOP  000326           333#
$DEVCT  000310           324#
$ENDAD  021520           311     4611    4625#
$ENO  = 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    1868#   1881    1884#   1899    1902#   1913    1916#   1933    1936#   1943    1946#
                         1967    1970#   1971    1974#   1988    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#   2891    2894#   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#
```



```
                         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#   3690    3693#   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#
$ENV    000320           329#
$ENVM   000321           330#    415     4609    4615
$FRROR= 000302           366#    405*
$ETABL  000320           328#
$FTEND  000338           340#    363
$FATAL  000302           321#    366     407
$HIBTS  000330           358#
$MAIL   000300           319#    359     363
$MBADR  000332           359#
$MSGAD  000314           326#
$MSGLG  000316           327#
$MSGTY  000300           320#    401#    406
$PASS   000306           323#    4606*
$PASTM  000336           361#
$SN   = 000151           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#
$SVPC = 000300           309#    314
$SWREG  000322           331#
$TESTN  000304           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#
```

CROSS REFERENCE TABLE -- USER SYMBOLS

```
                        3613*  3646*  3669*  3693*  3753*  3784*  3795*  3818*  3860*  3886*  4044*  412A*  4221*
                        4450*  4502*
$TSTM    000334          360#
$TSTNM= 000304          365#   404*
$UNIT    000312          325#
$UNITM   000340          362#
$USMP    000324          332#
.      = 021776          301#   302#   304#   309    310#   312#   314#   347    348#   350#   352#   369#   395#
                         472    610    765    900   1055   1192   1457   170A   1737   1756   1798   3701   3A00
                        3828   3846   3866   4354   4369   4409   4594   4601#  4636#  4659#
.$X    = 000330          347#   352
```

. ABS. 021776 000

```
% 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)
```