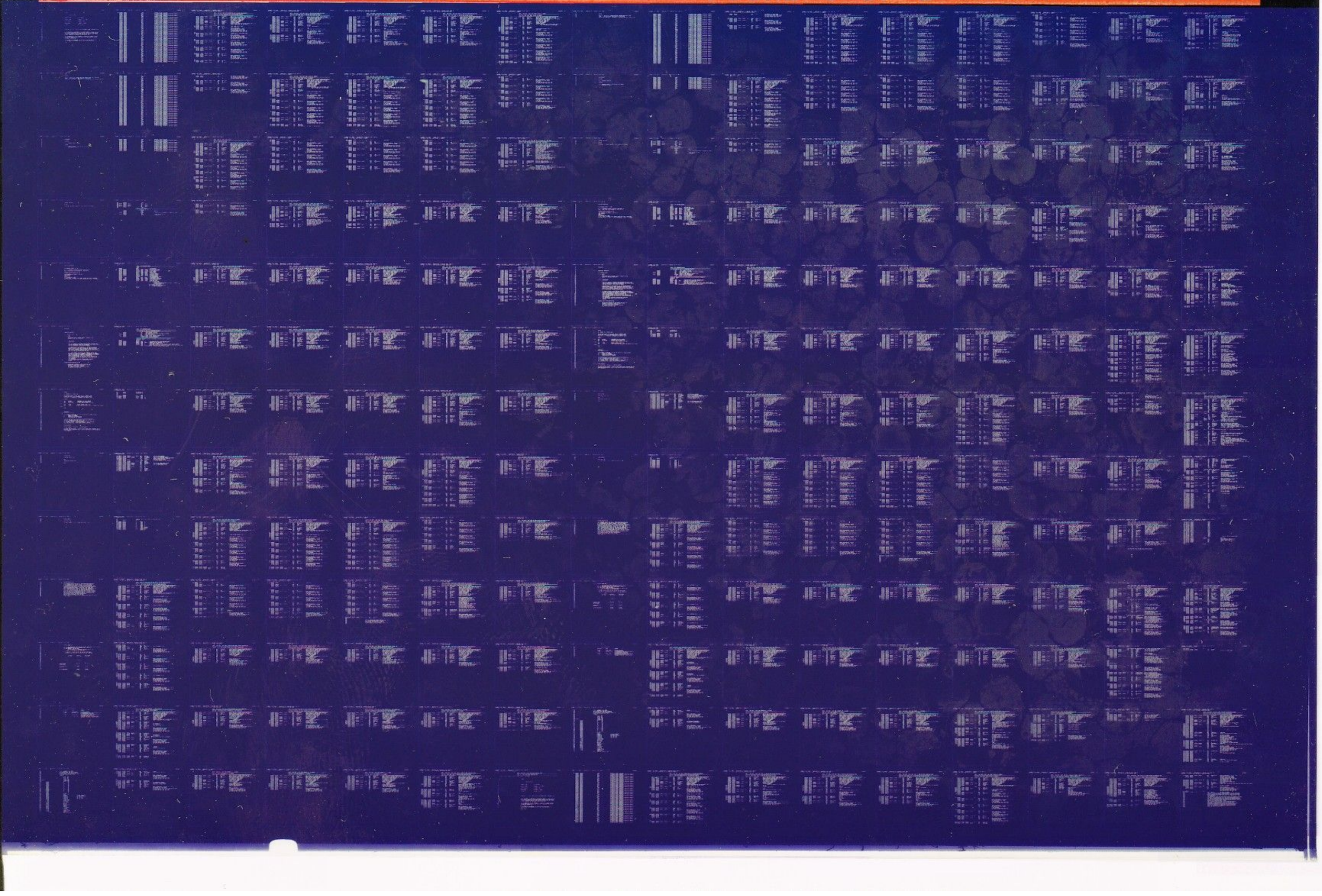
KD11-Z

11/44 TRAPS
CKKABD0

AH-F623D-MC
FICHE 1 OF 2

d13130



KD11-Z

11/44 TRAPS
CKKABDO

AH-F623D-MC
FICHE 2 OF 2

```
.REM %

        IDENTIFICATION

PRODUCT CODE:           AC-F621D-MC
PRODUCT NAME:           CKKABD0 11/44 TRAPS
DATE CREATED:           OCTOBER, 1981
MAINTAINER:             DIAGNOSTIC GROUP
AUTHOR:                 DAN MILLEVILLE
```

```
34                          1.      ABSTRACT
35
36                                  THIS IS A TEST OF ALL OPERATIONS AND INSTRUCTIONS THAT CAUSE
37                                  TRAPS.  ALSO TESTED ARE TRAP OVERFLOW CONDITIONS,
38                                  ODDITIES OF REGISTER 6, INTERRUPTS , THE RESET AND WAIT INSTRUCTIONS.
```

```
                        2.      REQUIREMENTS
                        2.1     EQUIPMENT
                                11/44 STANDARD COMPUTER
                        2.2     STORAGE
                                2.2.1 PROGRAM STORAGE - THE ROUTINE USES MEMORY
                                      FROM 0000 TO 17600.
```

```
                        3.      LOADING PROCEDURE

                        3.1     METHOD

                                PROCEDURE FOR NORMAL ABSOLUTE TAPES SHOULD BE FOLLOWED.
```

```
                        4.      STARTING PROCEDURE


                                THE PROGRAM STARTS  AT 200.

                                IF IT IS DESIRED TO RESET THE PASS COUNT BACK TO
                                ZERO ; THEN START THIS PROGRAM AT LOCATION 210

                        4.1     PROGRAM AND/OR OPERATOR ACTION

                                LOAD PROGRAM INTO MEMORY. (BOTTOM 4K)
                                LOAD ADDRESS.
                                START.
                                THE PROGRAM WILL LOOP.
                                IT WILL PRINT ''CKKABD0 11/44 TRAPS'' AFTER THE FIRST ITERATION
                                AND THEN PRINTS IT EVERY 15 TIMES(APROXIMATELY EVERY 15 SECONDS)
```

```
                        5.      OPERATION


                        5.1     SUBROUTINE ABSTRACTS

                                5.1.1 BEGIN AT 200

                                5.1.2  SCOPE
                                       -----
                                       IF A SCOPE LOOP IS NEEDED INSERT A BRANCH AS THE
                                       COMMENT TO THE HALT EXPLAINS.


                                5.1.3   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 PRINCIPLE OF THIS ROUTINE IS: THE VECTOR ENTRANCE
                                        ADDRESS POINTS TO THE NEXT SEQUENTIAL WORD WHICH WILL CON-
                                        TAIN 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
                                        WHERE THE PROGRAM WAS. WHEN THE INTERRUPT OR
                                        TRAP OCCURRED; MEMORY AS SPECIFIED BY R6 CONTAINS THE
                                        PC OF THE INSTRUCTION FOLLOWING THE INSTRUCTION WHERE THE
                                        TRAP OCCURRED.
                                        THE CONTENTS OF LOCATION '$TESTN'(304) CONTAINS
                                        THE TEST NUMBER THAT IT WAS DOING BEFORE IT
                                        TRAPPED.

                        5.2     PROGRAM AND/OR OPERATOR ACTION

                                5.2.1  LOADING AND STARTING AT 200 STARTS THE TEST. IF
                                       AN ERROR IS DETECTED, THERE WILL BE A HALT.
                                        NOTE:IF A SCOPE LOOP IS NEEDED
                                       THE COMMENT SECTION OF THE HALT EXPLAINS
                                        HOW TO UTILIZE THIS LOOP.
```

```
                        6.      ERRORS

                        6.1     ALL ERRORS WILL CAUSE A HALT.

                        6.1.1   THE PROGRAM CHECKS TO SEE THAT THE P.C. DOESN'T JUMP
                                WITHIN THE TESTS, BY A SEQUENCE COUNT CALLED '$TSTN'
                                THIS TEST IS A SEQUENCIAL INCREMENT AND COMPARE COUNT.

                                EXAMPLE

                        TSTA:   INC     $TSTNM          ;INCREMENT THE TEST NUMBER
                                CMP     #A,$TSTNM       ;COMPARE FOR THE RIGHT TEST
                                BNE     TSTA+1-12       ;IF NOT CORRECT BRANCH TO A HALT
                        **OR:
                                BEQ     1000$           ;BRANCH AROUND JUMP IF OK
                                JMP     TSTA+1-12       ;JUMP (USED FOR LONG TESTS THE BRANCH CAN'T REACH)
                        1000$:
                                ----
                                CODE



                                IMPORTANT
                        ******************************

                                IF AN ERROR IS DETECTED ;IT COULD BE BECAUSE OF THREE REASONS.
                                A)      WRONG TEST NUMBER
                                B)      ERROR IN THE PRESENT TEST.
                                C)      POWER MONITOR BIT PROBLEM.

                                //////////////////////////////////////////////////////////
                                THE TEST SEQUENCE LOCATION "TESTN" SHOULD BE CHECKED FIRST
                                TO SEE IF IT MATCHES THE PRESENT TEST.
                                IF IT DOESN'T MATCH ; THEN THE CONTENTS OF THIS LOCATION
                                TELL YOU WHICH TEST IT WAS DOING BEFORE IT HALTED.
                                //////////////////////////////////////////////////////////

                        6.2     ERROR RECOVERY

                                ON TRAP ERRORS - RESTART AT STARTING ADDRESS

                                ON POWER MONITOR ERRORS - FIND OUT WHAT POWER SUPPLY PROBLEM (OR CPU
                                ERROR REGISTER PROBLEM) THERE IS AND OBTAIN REPAIR BEFORE RERUNNING
                                THIS DIAGNOSTIC
```

```
                        7.      RESTRICTIONS
                        7.1     STARTING RESTRICTION
                                NONE
                        7.2     OPERATIONAL RESTRICTION
                                NONE
```

```
168
169
170
171
172
```

8. MISCELLANEOUS

8.1 EXECUTION TIME

1ST PASS APPROX. 2 SEC., THEREAFTER EVERY 15 SEC

```
                9.      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 RESERVED 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.
                        THE RTI AND RTT INSTRUCTIONS ARE CHECKED. STACK OVERFLOW IS
                        ALSO CHECKED FOR ALL THE TRAP INSTRUCTIONS.
                        SPECIAL CHECKS ARE MADE TO SEE IF BUS
                        ERROR TRAPS OCCUR ON NON-EXISTENT MEMORY.PIRQ TRAPS ARE CHECKED
                        AT ALL LEVELS.
```

```
                10.0    RUNNING UNDER APT

                        THE EXECUTION TIMES PROVIDED IN THE APT SCRIPT THAT FOLLOWS
                ARE FOR EXECUTION WITH A 11/44 PROCESSOR, CACHE,
                16K CORE MEMORY, AND 300 BAUD.
                        THE FOLLOWING IS A PROGRAM LOAD FILE USED BY APT:
                1. E TABLE 'A' IS USED FOR APT DUMP MODE.

                2. E TABLE 'B' IS USED FOR APT QV AND RUN TIME MODES.
                   $ENVM=040 INDICATES THAT TYPEOUTS WILL BE SUPRESSED.


                        1ST PASS        LONGEST         ADDITIONAL
                        RUN TIME        TEST TIME       RUN TIME
                        5                   5               0


                ......                  E TABLES        .......

                                        A               B
                E-MODE/S-MODE           000/000         040/001
                ($ENVM/$ENV)

                SWITCH REGISTER 1       000000          000000
                ($SWREG)

                SWITCH REGISTER 2       000000          000000
                CPU TYPE/OPTIONS        00/0000         00/0000
```

```
                        11.0            REVISION HISTORY

                                REVISION        DATE            COMMENT

                                CKKABA          MARCH 1979      ORIGINAL RELEASE
                                CKKABB          NOVEMBER 1980   FIX DIAGNOSTIC HALT WITH CIS SWITCH IN
                                                                THE MAINTENANCE POSITION
                                CKKABC          APRIL 1981      INCLUSION OF CPU ERROR REGISTER BIT 0
                                                                CHECKING BEFORE EACH TEST.
                                CKKABD          OCTOBER 1981    CHECKING THAT WITH PIRQ AND UNIBUS
                                                                INTERRUPTS PENDING, UNIBUS SERVICE IS
                                                                1ST.
                        %
```

```
                        .TITLE  CKKABD0 11/44 TRAPS
                        ;ALL INSTRUCTIONS THAT ARE RESERVED
                        ;SHOULD TRAP TO LOCATION 10, AND THE
                        ;PC THAT POINTS TO THE TRAPPING INSTRUCTION
                        ;SHOULD BE PLACED ON THE STACK
                        ;LISTING
                                .LIST   ME
                                .NLIST  MC
                                .NLIST  MD
                                .NLIST  CND
000000                          .ENABLE ABS
                                .ENABLE AMA
000000                          R0      =%0
000001                          R1      =%1
000001                          LAST    =%1
000002                          R2      =%2
000003                          R3      =%3
000003                          TAP     =%3
000004                          R4      =%4
000005                          R5      =%5
000005                          FIRST   =%5
000006                          R6      =%6
000006                          SP      =%6
000007                          R7      =%7
000007                          PC      =%7
000000                          HLT=HALT
000003                          TRT=3
000004                          RTRAP5=4                ;ILLEGAL ADDRESSES
000014                          RTRAP4=14               ;FOR TRACE TRAP
000030                          RTRAP3=30               ;FOR EMULATOR TRAP
000020                          RTRAP2=20               ;FOR IOT TRAP
000034                          RTRAP1=34               ;FOR TRAP INST
177564                          TTCSR=177564
177560                          TRCSR=177560
177564                          TPS=177564
177566                          TPB=177566
000240                          BELL=240
000240                          NOP=240
177776                          STATUS=177776
000010                          TRAPA=10
000010                          RTRAP=10
004700                          ILLA=004700
000100                          ILLB=100
177776                          CC=177776
177766                          CPUERP=177766
177413                          CERMSK=177413
177776                          PSW=177776
                        .MCALL  .$APTHDR,.$APTBLS,.$ACT11
```

```
382 000000  000002                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000002  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000004  000006                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000006  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000010  000012                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000012  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000014  000016                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000016  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000020  000022                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000022  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000024  000026                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000026  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000030  000032                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000032  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000034  000036                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000036  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000040  000042                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000042  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000044  000046                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000046  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000050  000052                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000052  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000054  000056                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000056  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000060  000062                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000062  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000064  000066                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000066  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000070  000072                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000072  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000074  000076                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000076  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000100  000102                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000102  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000104  000106                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000106  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000110  000112                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000112  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000114  000116                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000116  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000120  000122                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000122  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000124  000126                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000126  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000130  000132                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000132  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000134  000136                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000136  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000140  000142                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000142  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000144  000146                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000146  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000150  000152                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000152  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000154  000156                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
    000156  000000                  HALT             ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000160  000162                  .WORD   .+2      ;ADDRESS OF NEXT LOCATION
```

```
000162  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000164  000166                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000166  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000170  000172                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000172  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000174  000176                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000176  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000200  000202                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000202  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000204  000206                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000206  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000210  000212                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000212  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000214  000216                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000216  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000220  000222                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000222  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000224  000226                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000226  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000230  000232                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000232  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000234  000236                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000236  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000240  000242                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000242  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000244  000246                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000246  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000250  000252                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000252  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000254  000256                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000256  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000260  000262                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000262  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000264  000266                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000266  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000270  000272                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000272  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000274  000276                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000276  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000300  000302                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000302  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000304  000306                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000306  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000310  000312                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000312  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000314  000316                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000316  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000320  000322                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000322  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000324  000326                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000326  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000330  000332                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000332  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000334  000336                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000336  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000340  000342                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000342  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
```

```
000344  000346                  .WORD   .+2            ;ADDRESS OF NEXT LOCATION
000346  000000                  HALT                   ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000350  000352                  .WORD   .+2            ;ADDRESS OF NEXT LOCATION
000352  000000                  HALT                   ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRE
000354  000356                  .WORD   .+2            ;ADDRESS OF NEXT LOCATION
000356  000000                  HALT                   ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000360  000362                  .WORD   .+2            ;ADDRESS OF NEXT LOCATION
000362  000000                  HALT                   ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000364  000366                  .WORD   .+2            ;ADDRESS OF NEXT LOCATION
000366  000000                  HALT                   ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000370  000372                  .WORD   .+2            ;ADDRESS OF NEXT LOCATION
000372  000000                  HALT                   ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000374  000376                  .WORD   .+2            ;ADDRESS OF NEXT LOCATION
000376  000000                  HALT                   ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
```

```
                000200                          .=200
000200  000137  000510                          JMP     BEGIN
                000210                          .=210
000210  005037  000306                          CLR     $PASS
000214  000137  000510                          JMP     BEGIN
                000300                          .=300
                                        .SBTTL  ACT11 HOOKS
                                        ;*****************************************************************************
                                        ;HOOKS REQUIRED BY ACT11
                000300                          $SVPC=.                 ;SAVE PC
                000046                          .=46
        000046  023550                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
                000052                          .=52
        000052  000000                          .WORD   0               ;;2)SET LOC.52 TO ZERO
                000300                          .=$SVPC                 ;; RESTORE PC
```

```
393                                     .SBTTL  APT MAILBOX-ETABLE
                                        ;******************************************************************************
                                        .EVEN
000300                                  $MAIL:                          ;;APT MAILBOX
000300  000000                          $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
000302  000000                          $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
000304  000000                          $TESTN: .WORD   ATESTN          ;;TEST NUMBER
000306  000000                          $PASS:  .WORD   APASS           ;;PASS COUNT
000310  000000                          $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
000312  000000                          $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
000314  000000                          $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
000316  000000                          $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
000320                                  $ETABLE:                        ;;APT ENVIRONMENT TABLE
000320     000                          $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
000321     000                          $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
000322  000000                          $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
000324  000000                          $USWR:  .WORD   AUSWR           ;;USER SWITCHES
000326  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
```

```
    395                                 .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
                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                  $HIBTS: .WORD   0         ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
        000332  000300                  $MBADR: .WORD   $MAIL     ;;ADDRESS OF APT MAILBOX (BITS 0-15)
        000334  000005                  $TSTM:  .WORD   5         ;;RUN TIM OF LONGEST TEST
        000336  000005                  $PASTM: .WORD   5         ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
        000340  000000                  $UNITM: .WORD   0         ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
        000342  000014                          .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```

APT PARAMETER BLOCK

```
    397           000304                  $TSTNM=$TESTN
    398           000302                  $ERROR=$FATAL
    399
    400           000500                          .=500
    401 000500    000000                  BUFF:   .WORD   0
    402 000502    000000                  RCPUER: .WORD   0
    403 000504    000250                  KTVEC:  .WORD   250
    404 000506    000252                  KTSTA:  .WORD   252
```

```
                                BEGIN:  MOV     #500,SP         ;SET UP STACK POINTER
                                        MOV     #-1,PASSPT      ;CLEAR THE ITERATION COUNTER
                                        CMP     42,$ENDAD       ;SEE IF TITLE MESSAGE SHOULD BE PRINTED
                                        BEQ     RESTRT          ;BRANCH AROUND MESSAGE PRINTING IF SO
                                        JSR     PC,PRTMSG       ;GO PRINT MESSAGE USING
                                        .WORD   TITLE           ;STARTING ADDRESS OF THIS MESSAGE
                                RESTRT: CLR     $MSGTY
                                        MOV     #500,R6
                                        MOV     #PWRDWN,24      ;SET UP THE POWER DOWN VECTOR
                                        MOV     #340,26         ;SET UP POWER DOWN PRIORITY
                                        MOV     #6,4            ;SET UP TRAP VECTORS 4 & 6.
                                        CLR     6
                                        MOV     #12,10
                                        CLR     12
                                        CLR     $TSTNM
                                        CLR     $ERROR
                                        MOV     #$MSGTY,R2
                                        BR      TST1
```

```
                                        ;SPECIAL CASE OF ODD;.EVEN .BYTE AND REGISTER 6
000000                                  HERE=0

000626  000000                          K1:     .WORD   0
000630  000000                          K2:     .WORD   0
000632  000000                          K3:     .WORD   0
000634  000000                          K4:     .WORD   0
000636  000000                          K5:     .WORD   0
000640  000000                          K6:     .WORD   0
000642  052525                          K7:     .WORD   052525
000644  052400                          K10:    .WORD   052400
000646  000000                          K11:    .WORD   0
000650  000000                          K12:    .WORD   0
```

```
                                        .SBTTL  TEST #1 - TEST AUTO INC AND DEC OF R6 FOR WORD AND BYTES
                                        ;*****************************************************************************
                                        ;TEST 1 - TEST AUTO INC AND DEC OF R6 FOR WORD AND BYTES
                                        ;*****************************************************************************
000652  032737  000001  177766  TST1:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
000660  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
000662  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
000670  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
000672  000000                          HALT                    ;CPU POWER BIT FOUND SET
000674  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
000702  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
000706  022737  000001  000304          CMP     #1,$TESTN       ;SEQUENCE ERROR?
000714  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
000716  000137  001220                  JMP     TST2-12         ;JUMP TO ERROR HALT ON SEQ ERROR
000722                          1000$:
437 000722  005006                      CLR     R6
438 000724  112637  000000              MOVB    (R6)+,HERE      ;SIX SHOULD INCREMENT BY TWO
439 000730  020627  000002              CMP     R6,#2
440 000734  001405                      BEQ     1$
000736  012737  000001  000302          MOV     #1,$FATAL       ;MOVE TO MAILBOX #  *******  1  *******
000744  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
000746  000000                          HALT                    ;R6 DID NOT AUTO INCREMENT BY TWO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764
441
442 000750  012706  001000          1$:     MOV     #1000,R6
443 000754  114627  000000              MOVB    -(R6),#HERE     ;SHOULD DECREMENT BY TWO
444 000760  020627  000776              CMP     R6,#776
445 000764  001405                      BEQ     2$
000766  012737  000002  000302          MOV     #2,$FATAL       ;MOVE TO MAILBOX #  *******  2  *******
000774  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
000776  000000                          HALT                    ;R6 DID NOT AUTO DECREMENT BY 2
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 750
446
447 001000  005006                  2$:     CLR     R6
448 001002  112626                          MOVB    (R6)+,(R6)+     ;DOUBLES AUTO INCREMENT OF R6
449 001004  020627  000004              CMP     R6,#4
450 001010  001405                      BEQ     3$
001012  012737  000003  000302          MOV     #3,$FATAL       ;MOVE TO MAILBOX #  *******  3  *******
001020  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001022  000000                          HALT                    ;WRONG AUTO INCREMENT OF R6
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 736
451
452 001024  005006                  3$:     CLR     R6
453 001026  005004                          CLR     R4
454 001030  122624                          CMPB    (R6)+,(R4)+     ;TEST INCREMENT OF R6
455 001032  020627  000002              CMP     R6,#2
456 001036  001405                      BEQ     4$
001040  012737  000004  000302          MOV     #4,$FATAL       ;MOVE TO MAILBOX #  *******  4  *******
001046  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001050  000000                          HALT                    ;WRONG INCREMENT OF R6
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 723
457
458 001052  005006                  4$:     CLR     R6
```

```
                                        CLR     R4
                                        CMPB    (R4)+,(R6)+     ;TEST INCREMENT OF R6
                                        CMP     R6,#2
                                        BEQ     5$
001066  012737  000005  000302          MOV     #5,$FATAL       ;MOVE TO MAILBOX #  *******  5  *******
001074  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001076  000000                          HALT                    ;WRONG INCREMENT OF R6
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 710

                                5$:     CLR     R6
                                        CLR     R4
                                        CMPB    (R6)+,(R4)+     ;TEST INCREMENT OF R4
                                        CMP     R4,#1
                                        BEQ     6$
001114  012737  000006  000302          MOV     #6,$FATAL       ;MOVE TO MAILBOX #  *******  6  *******
001122  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001124  000000                          HALT                    ;WRONG INCREMENT OF R4
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 675

                                6$:     CLR     R6
                                        CLR     R4
                                        CMPB    (R4)+,(R6)+     ;TEST INCREMENT OF R6
                                        CMP     R6,#2
                                        BEQ     7$
001142  012737  000007  000302          MOV     #7,$FATAL       ;MOVE TO MAILBOX #  *******  7  *******
001150  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001152  000000                          HALT                    ;WRONG INCREMENT OF R6
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 662

                                7$:     CLR     R6
                                        CLR     R4
                                        CMPB    (R4)+,(R6)+     ;TEST INCREMENT OF R4
                                        CMP     R4,#1
                                        BEQ     8$
001170  012737  000010  000302          MOV     #10,$FATAL      ;MOVE TO MAILBOX #  *******  10  *******
001176  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001200  000000                          HALT                    ;WRONG INCREMENT OF R4
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 647

001202  012706  001000          8$:     MOV     #1000,R6
001206  124627  000000                  CMPB    -(R6),#HERE     ;TEST DECREMENT OF R6
001212  022706  000776                  CMP     #776,R6
001216  001405                          BEQ     TST2
001220  012737  000011  000302          MOV     #11,$FATAL      ;MOVE TO MAILBOX #  *******  11  *******
001226  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001230  000000                          HALT                    ;WRONG DECREMENT OF R6,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 633
```

```
                                .SBTTL  TEST #2 - TEST TRANSFER OF .BYTE USING R6
                                ;*****************************************************************
                                ;TEST 2 - TEST TRANSFER OF .BYTE USING R6
                                ;*****************************************************************
001232  032737  000001  177766  TST2:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
001240  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
001242  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
001250  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001252  000000                         HALT                    ;CPU POWER BIT FOUND SET
001254  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
001262  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
001266  022737  000002  000304         CMP     #2,$TESTN       ;SEQUENCE ERROR?
001274  001402                         BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
001276  000137  001600                 JMP     TST3-12         ;JUMP TO ERROR HALT ON SEQ ERROR
001302                          1000$:
487 001302  012737  123456  000636         MOV     #123456,K5
488 001310  012737  050505  000626         MOV     #050505,K1
489 001316  012705  000626                 MOV     #K1,R5          ;R5=(050505)K1
490 001322  012706  000636                 MOV     #K5,R6          ;R6=(123456)K5
491 001326  112625                         MOVB    (R6)+,(R5)+     ;LOW .BYTE OF R6 TO R5
492 001330  022737  050456  000626         CMP     #050456,K1
493 001336  001405                         BEQ     1$
001340  012737  000012  000302         MOV     #12,$FATAL      ;MOVE TO MAILBOX #  *******  12  *******
001346  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001350  000000                         HALT                    ;FALSE TRANSFER OF .BYTE
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
494
495 001352  012737  123456  000636  1$:     MOV     #123456,K5
496 001360  012737  050505  000626         MOV     #050505,K1
497 001366  012705  000626                 MOV     #K1,R5          ;R5(050505)K1
498 001372  012706  000640                 MOV     #K6,R6          ;R6(123456)K5
499 001376  114625                         MOVB    -(R6),(R5)+     ;LOW .BYTE OF R6 TO R5 (DECREMENT)
500 001400  023727  000626  050456         CMP     K1,#050456
501 001406  001405                         BEQ     2$
001410  012737  000013  000302         MOV     #13,$FATAL      ;MOVE TO MAILBOX #  *******  13  *******
001416  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001420  000000                         HALT                    ;FALSE R6 .BYTE TRANSFER
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 727
502
503 001422  012737  123456  000626  2$:     MOV     #123456,K1
504 001430  012737  050505  000636         MOV     #050505,K5
505 001436  012705  000626                 MOV     #K1,R5          ;(123456)
506 001442  012706  000636                 MOV     #K5,R6          ;(050505)
507 001446  112526                         MOVB    (R5)+,(R6)+     ;LOW OF R5 TO LOW OF R6
508 001450  022737  050456  000636         CMP     #050456,K5
509 001456  001405                         BEQ     3$
001460  012737  000014  000302         MOV     #14,$FATAL      ;MOVE TO MAILBOX #  *******  14  *******
001466  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001470  000000                         HALT                    ;FALSE R6 .BYTE TRANSFER
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 703
510
511 001472  012737  123456  000626  3$:     MOV     #123456,K1
512 001500  012737  050505  000636         MOV     #050505,K5
513 001506  012705  000627                 MOV     #K1+1,R5        ;123456
```

TEST #2 - TEST TRANSFER OF .BYTE USING R6

```
514 001512  012706  000636                    MOV     #K5,R6          ;050505
515 001516  112526                            MOVB    (R5)+,(R6)+     ;HIGH OF R5 TO LOW OF R6
516 001520  023727  000636  050647            CMP     K5,#050647
517 001526  001405                            BEQ     4$
    001530  012737  000015  000302            MOV     #15,$FATAL      ;MOVE TO MAILBOX #  *******  15  *******
    001536  005212                            INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    001540  000000                            HALT                    ;FALSE R6 .BYTE TRANSFER
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 657
518
519 001542  012737  123456  000626  4$:       MOV     #123456,K1
520 001550  012737  050505  000636            MOV     #050505,K5
521 001556  012705  000627                    MOV     #K1+1,R5        ;R5-123456-ODD ADDRESS
522 001562  012706  000636                    MOV     #K5,R6          ;R6-050505--.EVEN ADDRESS
523 001566  112625                            MOVB    (R6)+,(R5)+     ;LOW OF R6 TO HIGH OF R5
524 001570  022737  042456  000626            CMP     #042456,K1
525 001576  001405                            BEQ     TST3
    001600  012737  000016  000302            MOV     #16,$FATAL      ;MOVE TO MAILBOX #  *******  16  *******
    001606  005212                            INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    001610  000000                            HALT                    ;FAILED LOW OF 6 TO HIGH OF 5,OR WRONG $TSTNM
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 633
```

TEST #3 - TEST BYTE OPERATION WITH SEQ ODD-EVEN ADDRESS

```
526                                     .SBTTL  TEST #3 - TEST BYTE OPERATION WITH SEQ ODD-EVEN ADDRESS
                                        ;*************************************************************************
                                        ;TEST 3 - TEST BYTE OPERATION WITH SEQ ODD-EVEN ADDRESS
                                        ;*************************************************************************
001612  032737  000001  177766  TST3:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
001620  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
001622  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
001630  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001632  000000                          HALT                    ;CPU POWER BIT FOUND SET
001634  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
001642  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
001646  022737  000003  000304          CMP     #3,$TESTN       ;SEQUENCE ERROR?
001654  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
001656  000137  002070                  JMP     TST4-12         ;JUMP TO ERROR HALT ON SEQ ERROR
001662                          1000$:
527 001662  123737  000642  000643          CMPB    K7,K7+1         ;SAME .WORD LOW TC HIGH
528 001670  001405                          BEQ     5$
001672  012737  000017  000302          MOV     #17,$FATAL      ;MOVE TO MAILBOX #  *******  17  *******
001700  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001702  000000                          HALT                    ;SHOULD COMPARE LOW TO HIGH
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 766
529
530 001704  123737  000643  000642  5$:     CMPB    K7+1,K7         ;COMPARE ODD TO .EVEN SAME .WORD
531 001712  001405                          BEQ     6$
001714  012737  000020  000302          MOV     #20,$FATAL      ;MOVE TO MAILBOX #  *******  20  *******
001722  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001724  000000                          HALT                    ;ODD TO .EVEN .BYTE FAILURE
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
532
533 001726  123737  000645  000642  6$:     CMPB    K10+1,K7        ;SEQUENTIAL .BYTES
534 001734  001405                          BEQ     7$
001736  012737  000021  000302          MOV     #21,$FATAL      ;MOVE TO MAILBOX #  *******  21  *******
001744  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001746  000000                          HALT                    ;ODD TO .EVEN FAILED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 744
535
536 001750  123737  000644  000640  7$:     CMPB    K10,K6
537 001756  001405                          BEQ     8$
001760  012737  000022  000302          MOV     #22,$FATAL      ;MOVE TO MAILBOX #  *******  22  *******
001766  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001770  000000                          HALT                    ;.EVEN TO EVEN FAILED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 733
538 001772  123737  000643  000645  8$:     CMPB    K7+1,K10+1
539 002000  001405                          BEQ     9$
002002  012737  000023  000302          MOV     #23,$FATAL      ;MOVE TO MAILBOX #  *******  23  *******
002010  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002012  000000                          HALT                    ;ODD TO ODD FAILED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 722
540
541 002014  123737  000644  000645  9$:     CMPB    K10,K10+1
542 002022  001005                          BNE     10$
002024  012737  000024  000302          MOV     #24,$FATAL      ;MOVE TO MAILBOX #  *******  24  *******
```

```
        002032  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        002034  000000                          HALT                    ;LOW TO HIGH IN SAME .WORD FAILED
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 711

    543
    544 002036  123737  000645  000645  10$:    CMPB    K10+1,K10+1
    545 002044  001405                          BEQ     11$
        002046  012737  000025  000302          MOV     #25,$FATAL      ;MOVE TO MAILBOX #  *******  25  *******
        002054  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        002056  000000                          HALT                    ;HIGH TO LOW IN SAME .WORD FAILED
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 700

    546
    547 002060  123737  000644  000643  11$:    CMPB    K10,K7+1
    548 002066  001005                          BNE     TST4
        002070  012737  000026  000302          MOV     #26,$FATAL      ;MOVE TO MAILBOX #  *******  26  *******
        002076  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        002100  000000                          HALT                    ;.EVEN TO ODD FAILED,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 667
```

```
549                                     .SBTTL  TEST #4 - TEST THE CC BITS
                                        ;******************************************************************************
                                        ;TEST 4 - TEST THE CC BITS
                                        ;******************************************************************************
002102  032737  000001  177766  TST4:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002110  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002112  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002120  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002122  000000                          HALT                    ;CPU POWER BIT FOUND SET
002124  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002132  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002136  022737  000004  000304          CMP     #4,$TESTN       ;SEQUENCE ERROR?
002144  001062                          BNE     TST5-12         ;BRANCH TO ERROR HALT ON SEQ ERROR
550 002146  000277                      SCC                     ;SET STATUS
551 002150  005037  177776              CLR     STATUS          ;CLEAR STATUS
552 002154  103005                      BCC     1$
002156  012737  000027  000302          MOV     #27,$FATAL      ;MOVE TO MAILBOX #  *******  27  *******
002164  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002166  000000                          HALT                    ;C NOT CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 766
553 002170                      1$:
002170  102005                          BVC     2$
002172  012737  000030  000302          MOV     #30,$FATAL      ;MOVE TO MAILBOX #  *******  30  *******
002200  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002202  000000                          HALT                    ;V NOT CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
554 002204                      2$:
002204  001005                          BNE     3$
002206  012737  000031  000302          MOV     #31,$FATAL      ;MOVE TO MAILBOX #  *******  31  *******
002214  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002216  000000                          HALT                    ;Z NOT CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 752
555 002220                      3$:
002220  100005                          BPL     4$
002222  012737  000032  000302          MOV     #32,$FATAL      ;MOVE TO MAILBOX #  *******  32  *******
002230  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002232  000000                          HALT                    ;N NOT CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 744
556 002234  000257              4$:     CCC                     ;CLEAR CONDITION CODES
557 002236  052737  000017  177776      BIS     #17,STATUS      ;SET STATUS TO ONES
558
559 002244  103405                      BCS     5$
002246  012737  000033  000302          MOV     #33,$FATAL      ;MOVE TO MAILBOX #  *******  33  *******
002254  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002256  000000                          HALT                    ;C NOT SET
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
560 002260                      5$:
002260  102405                          BVS     6$
002262  012737  000034  000302          MOV     #34,$FATAL      ;MOVE TO MAILBOX #  *******  34  *******
002270  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002272  000000                          HALT                    ;V NOT SET
                                                                ;TO SCOPE REPLACE HALT WITH 240
```

```
                                                                ;AND REPLACE NEXT INST WITH 724

561 002274                                      6$:
    002274  001405                                     BEQ     7$
    002276  012737  000035  000302                     MOV     #35,$FATAL      ;MOVE TO MAILBOX #  *******  35  *******
    002304  005212                                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002306  000000                                     HALT                    ;Z NOT SET
                                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                                ;AND REPLACE NEXT INST WITH 716

562 002310                                      7$:
    002310  100405                                     BMI     TST5
    002312  012737  000036  000302                     MOV     #36,$FATAL      ;MOVE TO MAILBOX #  *******  36  *******
    002320  005212                                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002322  000000                                     HALT                    ;N NOT SET,OR WRONG $TSTNM
                                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                                ;AND REPLACE NEXT INST WITH 710
```

TEST #5 - TEST THAT A TRAP OCCURS ON A RESERVED INS

```
                                        .SBTTL  TEST #5 - TEST THAT A TRAP OCCURS ON A RESERVED INS
                                        ;******************************************************************************
                                        ;TEST 5 - TEST THAT A TRAP OCCURS ON A RESERVED INS
                                        ;******************************************************************************
002324  032737  000001  177766  TST5:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002332  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002334  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002342  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002344  000000                          HALT                    ;CPU POWER BIT FOUND SET
002346  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002354  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002360  022737  000005  000304          CMP     #5,$TESTN       ;SEQUENCE ERROR?
002366  001006                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
564 002370  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
565 002374  012737  002416  000010      MOV     #2$,RTRAP       ;RETURN LOCATION
566 002402  000010                      TRAPA                   ;RESERVED INSTRUCTION, SHOULD TRAP
567 002404                          1$:
002404  012737  000037  000302          MOV     #37,$FATAL      ;MOVE TO MAILBOX #  *******  37  *******
002412  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002414  000000                          HALT                    ;RESERVE INSTRUCTION DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764
568 002416                          2$:
```

```
569                                     .SBTTL  TEST #6 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************************
                                        ;TEST 6 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************************
002416  032737  000001  177766  TST6:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002424  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002426  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002434  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002436  000000                          HALT                    ;CPU POWER BIT FOUND SET
002440  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002446  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002452  022737  000006  000304          CMP     #6,$TESTN       ;SEQUENCE ERROR?
002460  001011                          BNE     TST7-12         ;BRANCH TO ERROR HALT ON SEQ ERROR
570 002462  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
571 002466  012737  002476  000010      MOV     #1$,RTRAP       ;RETURN POINTER
572 002474  000010                      TRAPA                   ;RESERVED INSTRUCTION
573 002476  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
574 002502  001405                      BEQ     TST7
    002504  012737  000040  000302      MOV     #40,$FATAL      ;MOVE TO MAILBOX #  *******  40  *******
    002512  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002514  000000                      HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

TEST #7 - TEST THAT PROPER P.C. IS SAVED

```
                                .SBTTL  TEST #7 - TEST THAT PROPER P.C. IS SAVED
                                ;*****************************************************************
                                ;TEST 7 - TEST THAT PROPER P.C. IS SAVED
                                ;*****************************************************************
002516  0?2737  000001  177766  TST7:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002524  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002526  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002534  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002536  000000                          HALT                    ;CPU POWER BIT FOUND SET
002540  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002546  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002552  022737  000007  000304          CMP     #7,$TESTN       ;SEQUENCE ERROR?
002560  001012                          BNE     TST10-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
576 002562  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
577 002566  012737  002576  000010      MOV     #1$,RTRAP       ;RETURN FROM TRAP POINTER
578 002574  000010                      TRAPA                   ;TRAP ON THIS INSTRUCTION
579 002576  022737  002576  000474  1$:     CMP     #1$,BUFF-4      ;CHECK FOR INCREMENTED P.C.
580 002604  001405                          BEQ     TST10
    002606  012737  000041  000302          MOV     #41,$FATAL      ;MOVE TO MAILBOX #  *******  41  *******
    002614  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002616  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 760
```

```
                                        .SBTTL  TEST #10 - TEST THAT 'OLD'' CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 10 - TEST THAT 'OLD'' CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
002620  032737  000001  177766  TST10:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002626  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002630  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002636  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002640  000000                          HALT                    ;CPU POWER BIT FOUND SET
002642  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002650  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002654  022737  000010  000304          CMP     #10,$TESTN      ;SEQUENCE ERROR?
002662  001046                          BNE     TST11-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
582 002664  012706  000500              MOV     #BUFF,SP        ;SET UP
583 002670  012737  002706  000010      MOV     #1$,RTRAP       ;SET UP
584 002676  005037  177776              CLR     CC              ;CLEAR CC AND PRICRITY
585 002702  000257                      CCC
586 002704  000010                      TRAPA                   ;TRAP
587 002706  023727  000476  000000  1$:  CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
588 002714  001405                      BEQ     2$
    002716  012737  000042  000302      MOV     #42,$FATAL      ;MOVE TO MAILBOX #  *******  42  *******
    002724  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002726  000000                      HALT                    ;INCORRECT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
589 002730  012706  000500          2$:  MOV     #BUFF,SP        ;SET UP
590 002734  012737  002754  000010      MOV     #3$,RTRAP       ;SET UP
591 002742  012737  000357  177776      MOV     #357,CC         ;SET PRIORITY
592 002750  000277                      SCC                     ;SET CC
593 002752  000010                      TRAPA                   ;TRAP
594 002754  023727  000476  000357  3$:  CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
595 002762  000405                      BR      4$              ;BRANCH OVER ERROR #177
596 002764  012737  000043  000302      MOV     #43,$FATAL      ;MOVE TO MAILBOX #  *******  43  *******
    002772  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002774  000000                      HALT                    ;THIS ERROR RESERVED FOR PWR MON BIT ERROR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
597 002776                          4$:
    002776  001405                      BEQ     TST11
    003000  012737  000044  000302      MOV     #44,$FATAL      ;MOVE TO MAILBOX #  *******  44  *******
    003006  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003010  000000                      HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 724
```

```
                                        .SBTTL  TEST #11 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*************************************************************************
                                        ;TEST 11 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*************************************************************************
003012  032737  000001  177766  TST11:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003020  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
003022  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
003030  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003032  000000                          HALT                    ;CPU POWER BIT FOUND SET
003034  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
003042  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
003046  022737  000011  000304          CMP     #11,$TESTN      ;SEQUENCE ERROR?
003054  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
003056  000137  003324                  JMP     12$             ;JUMP TO ERROR HALT ON SEQ ERROR
003062                          1000$:
599 003062  012706  000500              MOV     #BUFF,SP
600 003066  012737  003102  000010      MOV     #1$,RTRAP
601 003074  005037  000012              CLR     RTRAP+2         ;CLEAR FUTURE PRIORITY AND CC
602 003100  000010                       RAPA
603 003102                          1$:                         ;TEST FOR "C" CLEARED
604 003102  100005                      BPL     2$
003104  012737  000045  000302          MOV     #45,$FATAL      ;MOVE TO MAILBOX #  *******  45  *******
003112  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003114  000000                          HALT                    ;N NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
605 003116                          2$:
003116  001005                          BNE     3$
003120  012737  000046  000302          MOV     #46,$FATAL      ;MOVE TO MAILBOX #  *******  46  *******
003126  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003130  000000                          HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
606 003132                          3$:
003132  102005                          BVC     4$
003134  012737  000047  000302          MOV     #47,$FATAL      ;MOVE TO MAILBOX #  *******  47  *******
003142  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003144  000000                          HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
607 003146                          4$:
003146  103005                          BCC     5$
003150  012737  000050  000302          MOV     #50,$FATAL      ;MOVE TO MAILBOX #  *******  50  *******
003156  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003160  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
608 003162  032737  000340  177776  5$:  BIT     #340,CC         ;TEST PRIORITY
609 003170  001405                      BEQ     6$
003172  012737  000051  000302          MOV     #51,$FATAL      ;MOVE TO MAILBOX #  *******  51  *******
003200  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003202  000000                          HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 726
610 003204  012706  000500          6$:  MOV     #BUFF,SP
611 003210  012737  003226  000010      MOV     #7$,RTRAP
612 003216  012737  000357  000012      MOV     #357,RTRAP+2    ;SET NEW "CC" AND PRIORITY
```

```
613 003224  000010                        TRAPA                   ;TRAP HERE
614 003226                          7$:
    003226  100405                        BMI     8$
    003230  012737  000052  000302        MOV     #52,$FATAL      ;MOVE TO MAILBOX #  *******  52  *******
    003236  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003240  000000                        HALT                    ;N NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 707

615 003242                          8$:
    003242  001405                        BEQ     9$
    003244  012737  0000[illegible]  000302        MOV     #53,$FATAL      ;MOVE TO MAILBOX #  *******  53  *******
    003252  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003254  000000                        HALT                    ;Z NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 701

616 003256                          9$:
    003256  102405                        BVS     10$
    003260  012737  000054  000302        MOV     #54,$FATAL      ;MOVE TO MAILBOX #  *******  54  *******
    003266  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003270  000000                        HALT                    ;V NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 673

617 003272                          10$:
    003272  103405                        BCS     11$
    003274  012737  000055  000302        MOV     #55,$FATAL      ;MOVE TO MAILBOX #  *******  55  *******
    003302  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003304  000000                        HALT                    ;C NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 665

618 003306  013706  177776          11$:  MOV     CC,SP
619 003312  042706  000017                BIC     #17,SP
620 003316  022706  000340                CMP     #340,SP
621 003322  001405                        BEQ     13$
    003324                          12$:
    003324  012737  000056  000302        MOV     #56,$FATAL      ;MOVE TO MAILBOX #  *******  56  *******
    003332  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003334  000000                        HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 651

622 003336  012737  000012  000010  13$:  MOV     #12,10
623 003344  005037  000012                CLR     12
```

TEST #12 - TEST THAT A TRAP OCCURS FOR A "TRAP" INSTRUCTION

```
624                                             .SBTTL  TEST #12 - TEST THAT A TRAP OCCURS FOR A "TRAP" INSTRUCTION
                                                ;*****************************************************************
                                                ;TEST 12 - TEST THAT A TRAP OCCURS FOR A "TRAP" INSTRUCTION
                                                ;*****************************************************************
     003350  032737  000001  177766  TST12:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     003356  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     003360  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     003366  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     003370  000000                          HALT                    ;CPU POWER BIT FOUND SET
     003372  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     003400  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     003404  022737  000012  000304          CMP     #12,$TESTN      ;SEQUENCE ERROR?
     003412  001013                          BNE     TST13-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
625  003414  012737  000012  000010          MOV     #12,10
626  003422  005037  000012                  CLR     12
627  003426  012706  000500                  MOV     #BUFF,SP        ;STACK POINTER SETUP
628  003432  012737  003454  000034          MOV     #1$,RTRAP1      ;RETURN LOCATION
629  003440  104400                          TRAP                    ;RESERVED INSTRUCTION, SHOULD TRAP
630  003442  012737  000057  000302          MOV     #57,$FATAL      ;MOVE TO MAILBOX #  *******  57  *******
     003450  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     003452  000000                          HALT                    ;TRAP DIDN'T TRAP,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 757

631  003454                             1$:
```

```
                                .SBTTL  TEST #13 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                ;*************************************************************************
                                ;TEST 13 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                ;*************************************************************************
003454  032737  000001  177766  TST13:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003462  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
003464  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
003472  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003474  000000                         HALT                    ;CPU POWER BIT FOUND SET
003476  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
003504  005237  000304          100$:  INC     $TESTN          ;UPDATE TEST NUMBER
003510  022737  000013  000304          CMP     #13,$TESTN      ;SEQUENCE ERROR?
003516  001011                         BNE     TST14-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
633 003520  012706  000500             MOV     #BUFF,SP        ;STACK POINTER SETUP
634 003524  012737  003534  000034     MOV     #1$,RTRAP1      ;RETURN POINTER
635 003532  104400                     TRAP                    ;RESERVED INSTRUCTION
636 003534  020627  000474          1$:    CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
637 003540  001405                     BEQ     TST14
003542  012737  000060  000302          MOV     #60,$FATAL      ;MOVE TO MAILBOX #  *******  60  *******
003550  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003552  000000                         HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

```
638                                     .SBTTL  TEST #14 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
                                        ;TEST 14 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
003554  032737  000001  177766  1ST14:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003562  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
003564  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
003572  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003574  000000                          HALT                    ;CPU POWER BIT FOUND SET
003576  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
003604  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
003610  022737  000014  000304          CMP     #14,$TESTN      ;SEQUENCE ERROR?
003616  001012                          BNE     TST15-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
639 003620  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
640 003624  012737  003634  000034      MOV     #1$,RTRAP1      ;RETURN FROM TRAP POINTER
641 003632  104400                      TRAP                    ;TRAP ON THIS INSTRUCTION
642 003634  022737  003634  000474  1$:     CMP     #1$,BUFF-4      ;CHECK INCREMENTED P.C.
643 003642  001405                          BEQ     TST15
003644  012737  000061  000302          MOV     #61,$FATAL      ;MOVE TO MAILBOX #  *******  61  *******
003652  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003654  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
```

```
644                                     .SBTTL  TEST #15 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 15 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
003656  032737  000001  177766  TST15:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003664  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
003666  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
003674  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003676  000000                          HALT                    ;CPU POWER BIT FOUND SET
003700  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
003706  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
003712  022737  000015  000304          CMP     #15,$TESTN      ;SEQUENCE ERROR?
003720  001037                          BNE     TST16-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
645 003722  012706  000500              MOV     #BUFF,SP        ;SET UP
646 003726  012737  003744  000034      MOV     #1$,RTRAP1      ;SET UP
647 003734  005037  177776              CLR     CC              ;CLEAR CC AND PRICRITY
648 003740  000257                      CCC
649 003742  104400                      TRAP                    ;TRAP
650 003744  023727  000476  000000  1$:  CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
651 003752  001405                      BEQ     2$
003754  012737  000062  000302          MOV     #62,$FATAL      ;MOVE TO MAILBOX #  *******  62  *******
003762  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003764  000000                          HALT                    ;INCORRECT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
652 003766  012706  000500          2$:  MOV     #BUFF,SP        ;SET UP
653 003772  012737  004010  000034      MOV     #3$,RTRAP1      ;SET UP
554 004000  012737  000357  177776      MOV     #357,CC         ;SET PRIORITY
655 004006  104400                      TRAP                    ;SET CC
656 004010  023727  000476  000357  3$:  CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
657 004016  001405                      BEQ     TST16
004020  012737  000063  000302          MOV     #63,$FATAL      ;MOVE TO MAILBOX #  *******  63  *******
004026  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004030  000000                          HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 733
```

```
                                        .SBTTL  TEST #16 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;******************************************************************************
                                        ;TEST 16 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;******************************************************************************
 004032  032737  000001  177766 TST16:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 004040  001410                 BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 004042  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 004050  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 004052  000000                 HALT                    ;CPU POWER BIT FOUND SET
 004054  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 004062  005237  000304 100$:  INC     $TESTN          ;UPDATE TEST NUMBER
 004066  022737  000016  000304         CMP     #16,$TESTN      ;SEQUENCE ERROR?
 004074  001402                 BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
 004076  000137  004344                 JMP     TST17-12                ;JUMP TO ERROR HALT ON SEQ ERROR
 004102                         1000$:
 004102  012706  000500         MOV     #BUFF,SP
 004106  012737  004122  000034         MOV     #1$,RTRAP1
 004114  005037  000036         CLR     RTRAP1+2        ;CLEAR FUTURE PRIORITY AND CC
 004120  104400                 TRAP
 004122                         1$:                     ;TEST FOR "C" CLEARED
 004122  100005                 BPL     2$
 004124  012737  000064  000302         MOV     #64,$FATAL      ;MOVE TO MAILBOX #  *******  64  *******
 004132  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 004134  000000                 HALT                    ;C NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 761
 004136                         2$:
 004136  001005                 BNE     3$
 004140  012737  000065  000302         MOV     #65,$FATAL      ;MOVE TO MAILBOX #  *******  65  *******
 004146  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 004150  000000                 HALT                    ;Z NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 753
 004152                         3$:
 004152  102005                 BVC     4$
 004154  012737  000066  000302         MOV     #66,$FATAL      ;MOVE TO MAILBOX #  *******  66  *******
 004162  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 004164  000000                 HALT                    ;V NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 745
 004166                         4$:
 004166  103005                 BCC     5$
 004170  012737  000067  000302         MOV     #67,$FATAL      ;MOVE TO MAILBOX #  *******  67  *******
 004176  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 004200  000000                 HALT                    ;C NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 737
 004202  032737  000340  177776 5$:     BIT     #340,CC         ;TEST PRIORITY
 004210  001405                 BEQ     6$
 004212  012737  000070  000302         MOV     #70,$FATAL      ;MOVE TO MAILBOX #  *******  70  *******
 004220  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 004222  000000                 HALT                    ;PRIORITY NOT ZERO
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 726
 004224  012706  000500         6$:     MOV     #BUFF,SP
 004230  012737  004246  000034         MOV     #7$,RTRAP1
 004236  012737  000357  000036         MOV     #357,RTRAP1+2   ;SET NEW "CC" AND PRIORITY
```

TEST #16 - TEST THAT "NEW" STATUS IS CORRECT

```
673 004244  104400                        TRAP                    ;TRAP HERE
674 004246                          7$:
675 004246  100405                        BMI     8$
    004250  012737  000071  000302        MOV     #71,$FATAL      ;MOVE TO MAILBOX #  *******  71  *******
    004256  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    004260  000000                        HALT                    ;N NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 707
676 004262                          8$:
    004262  001405                        BEQ     9$
    004264  012737  000072  000302        MOV     #72,$FATAL      ;MOVE TO MAILBOX #  *******  72  *******
    004272  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    004274  000000                        HALT                    ;Z NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 701
677 004276                          9$:
    004276  102405                        BVS     10$
    004300  012737  000073  000302        MOV     #73,$FATAL      ;MOVE TO MAILBOX #  *******  73  *******
    004306  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    004310  000000                        HALT                    ;V NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 673
678 004312                          10$:
    004312  103405                        BCS     11$
    004314  012737  000074  000302        MOV     #74,$FATAL      ;MOVE TO MAILBOX #  *******  74  *******
    004322  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    004324  000000                        HALT                    ;C NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 665
679 004326  013706  177776          11$:  MOV     CC,SP
680 004332  042706  000017              BIC     #17,SP
681 004336  022706  000340              CMP     #340,SP
682 004342  001405                      BEQ     TST17
    004344  012737  000075  000302        MOV     #75,$FATAL      ;MOVE TO MAILBOX #  *******  75  *******
    004352  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    004354  000000                        HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 651
```

```
                                        .SBTTL  TEST #17 - TEST THAT ALL COMB "TRAP" WILL CAUSE A TRAP
                                        ;******************************************************************
                                        ;TEST 17 - TEST THAT ALL COMB "TRAP" WILL CAUSE A TRAP
                                        ;******************************************************************
004356  032737  000001  177766  TST17:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004364  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
004366  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
004374  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004376  000000                          HALT                    ;CPU POWER BIT FOUND SET
004400  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
004406  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
004412  022737  000017  000304          CMP     #17,$TESTN      ;SEQUENCE ERROR?
004420  001011                          BNE     3$              ;BRANCH TO ERROR HALT ON SEQ ERROR
684 004422  012737  104400  004442      MOV     #TRAP,2$        ;INITIALIZE BASE TRAP INSTRUCTION
685 004430  012737  004456  000034      MOV     #4$,34          ;RETURN FROM TRAP TO RA1
686 004436  012706  000500          1$:  MOV     #BUFF,SP        ;SET UP STACK POINTER
687 004442  104400                  2$:  TRAP                    ;TRAP INST WILL BE MODIFIED TO TRAP+377
688 004444                          3$:
004444  012737  000076  000302          MOV     #76,$FATAL      ;MOVE TO MAILBOX #  *******  76  *******
004452  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004454  000000                          HALT                    ;PREVIOUS INST FAILED TO TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
689 004456  005237  004442          4$:  INC     2$              ;INCREMENT TRAP INSTRUCTION
690 004462  022737  104777  004442      CMP     #104777,2$      ;TRAP+377 TO UPPER LIMIT
691 004470  103362                      BHIS    1$              ;HAVE WE TESTED ALL
692 004472  012737  000036  000034      MOV     #36,34
693 004500  005037  000036              CLR     36
```

TEST #20 - TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION

```
694                                     .SBTTL  TEST #20 - TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION
                                        ;*************************************************************************
                                        ;TEST 20 - TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION
                                        ;*************************************************************************
     004504  032737  000001  177766  TST20:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     004512  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     004514  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     004522  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     004524  000000                          HALT                    ;CPU POWER BIT FOUND SET
     004526  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     004534  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     004540  022737  000020  000304          CMP     #20,$TESTN      ;SEQUENCE ERROR?
     004546  001006                          BNE     TST21-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
695  004550  012706  000500                  MOV     #BUFF,SP        ;STACK POINTER SETUP
696  004554  012737  004576  000020          MOV     #1$,RTRAP2      ;RETURN LOCATION
697  004562  000004                          IOT                     ;RESERVE INSTRUCTION, SHOULD TRAP
698  004564  012737  000077  000302          MOV     #77,$FATAL      ;MOVE TO MAILBOX #  *******  77  *******
     004572  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     004574  000000                          HALT                    ;IOT DIDN'T TRAP,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 764

699  004576                             1$:
```

```
 700                                    .SBTTL  TEST #21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*************************************************************************
                                        ;TEST 21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*************************************************************************
     004576  032737  000001  177766  TST21:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     004604  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     004606  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     004614  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     004616  000000                          HALT                    ;CPU POWER BIT FOUND SET
     004620  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     004626  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     004632  022737  000021  000304          CMP     #21,$TESTN      ;SEQUENCE ERROR?
     004640  001011                          BNE     TST22-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 701 004642  012706  000500                  MOV     #BUFF,SP        ;STACK POINTER SETUP
 702 004646  012737  004656  000020          MOV     #1$,RTRAP2      ;RETURN POINTER
 703 004654  000004                          IOT                     ;RESERVED INSTRUCTION
 704 004656  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
 705 004662  001405                          BEQ     TST22
     004664  012737  000100  000302          MOV     #100,$FATAL     ;MOVE TO MAILBOX #  *******  100  *******
     004672  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     004674  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 761
```

```
706                                     .SBTTL  TEST #22 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
                                        ;TEST 22 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
004676  032737  000001  177766  TST22:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004704  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
004706  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
004714  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004716  000000                          HALT                    ;CPU POWER BIT FOUND SET
004720  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
004726  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
004732  022737  000022  000304          CMP     #22,$TESTN      ;SEQUENCE ERROR?
004740  001012                          BNE     TST23-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
707 004742  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
708 004746  012737  004756  000020      MOV     #1$,RTRAP2      ;RETURN FROM TRAP POINTER
709 004754  000004                      IOT                     ;TRAP ON THIS INSTRUCTION
710 004756  022737  004756  000474  1$:     CMP     #1$,BUFF-4      ;CHECK FOR INCREMENTED P.C.
711 004764  001405                          BEQ     TST23
004766  012737  000101  000302          MOV     #101,$FATAL     ;MOVE TO MAILBOX #  *******  101  *******
004774  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004776  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
```

```
                                .SBTTL  TEST #23 - TEST THAT 'OLD'' CC AND PRI ARE PLACED ON STACK
                                ;******************************************************************
                                ;TEST 23 - TEST THAT 'OLD'' CC AND PRI ARE PLACED ON STACK
                                ;******************************************************************
 005000  032737  000001  177766  TST23:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 005006  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 005010  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 005016  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 005020  000000                         HALT                    ;CPU POWER BIT FOUND SET
 005022  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 005030  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 005034  022737  000023  000304         CMP     #23,$TESTN      ;SEQUENCE ERROR?
 005042  001040                         BNE     TST24-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 005044  012706  000500                 MOV     #BUFF,SP        ;SET UP
 005050  012737  005066  000020         MOV     #1$,RTRAP2      ;SET UP
 005056  005037  177776                 CLR     CC              ;CLEAR CC AND PRICRITY
 005062  000257                         CCC
 005064  000004                         IOT                     ;TRAP
 005066  023727  000476  000000  1$:     CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
 005074  001405                         BEQ     2$
 005076  012737  000102  000302         MOV     #102,$FATAL     ;MOVE TO MAILBOX #  *******  102  *******
 005104  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 005106  000000                         HALT                    ;INCORRECT STATUS
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 755
 005110  012706  000500         2$:     MOV     #BUFF,SP        ;SET UP
 005114  012737  005134  000020         MOV     #3$,RTRAP2      ;SET UP
 005122  012737  000357  177776         MOV     #357,CC         ;SET PRIORITY
 005130  000277                         SCC                     ;SET CC
 005132  000004                         IOT                     ;TRAP
 005134  023727  000476  000357  3$:     CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
 005142  001405                         BEQ     TST24
 005144  012737  000103  000302         MOV     #103,$FATAL     ;MOVE TO MAILBOX #  *******  103  *******
 005152  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 005154  000000                         HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 732
```

TEST #24 - TEST THAT 'NEW' STATUS IS CORRECT

```
                                .SBTTL  TEST #24 - TEST THAT 'NEW' STATUS IS CORRECT
                                ;*****************************************************************************
                                ;TEST 24 - TEST THAT 'NEW' STATUS IS CORRECT
                                ;*****************************************************************************
005156  032737  000001  177766  TST24:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005164  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
005166  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
005174  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005176  000000                         HALT                    ;CPU POWER BIT FOUND SET
005200  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
005206  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
005212  022737  000024  000304         CMP     #24,$TESTN      ;SEQUENCE ERROR?
005220  001402                         BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
005222  000137  005470                 JMP     12$             ;JUMP TO ERROR HALT ON SEQ ERROR
005226                          1000$:
728 005226  012706  000500             MOV     #BUFF,SP
729 005232  012737  005246  000020     MOV     #1$,RTRAP2
730 005240  005037  000022             CLR     RTRAP2+2        ;CLEAR FUTURE PRIORITY AND CC
731 005244  000004                     IOT
732 005246                      1$:                             ;TEST FOR "C" CLEARED
733 005246  100005                     BPL     2$
005250  012737  000104  000302         MOV     #104,$FATAL     ;MOVE TO MAILBOX #  *******  104  *******
005256  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005260  000000                         HALT                    ;N NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
734 005262                      2$:
005262  001005                         BNE     3$
005264  012737  000105  000302         MOV     #105,$FATAL     ;MOVE TO MAILBOX #  *******  105  *******
005272  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005274  000000                         HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
735 005276                      3$:
005276  102005                         BVC     4$
005300  012737  000106  000302         MOV     #106,$FATAL     ;MOVE TO MAILBOX #  *******  106  *******
005306  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005310  000000                         HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
736 005312                      4$:
005312  103005                         BCC     5$
005314  012737  000107  000302         MOV     #107,$FATAL     ;MOVE TO MAILBOX #  *******  107  *******
005322  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005324  000000                         HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
737 005326  032737  000340  177776  5$:     BIT     #340,CC         ;TEST PRIORITY
738 005334  001405                         BEQ     6$
005336  012737  000110  000302         MOV     #110,$FATAL     ;MOVE TO MAILBOX #  *******  110  *******
005344  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005346  000000                         HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 726
739 005350  012706  000500         6$:     MOV     #BUFF,SP
740 005354  012737  005372  000020         MOV     #7$,RTRAP2
741 005362  012737  000357  000022         MOV     #357,RTRAP2+2   ;SET NEW "CC" AND PRIORITY
```

TEST #24 - TEST THAT 'NEW" STATUS IS CORRECT

```
    005370  000004                          IOT                     ;TRAP HERE
                                    7$:
    005372  100405                          BMI     8$
    005374  012737  000111  000302          MOV     #111,$FATAL     ;MOVE TO MAILBOX #  *******  111  *******
    005402  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005404  000000                          HALT                    ;N NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 707

                                    8$:
    005406  001405                          BEQ     9$
    005410  012737  000112  000302          MOV     #112,$FATAL     ;MOVE TO MAILBOX #  *******  112  *******
    005416  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005420  000000                          HALT                    ;Z NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 701

                                    9$:
    005422  102405                          BVS     10$
    005424  012737  000113  000302          MOV     #113,$FATAL     ;MOVE TO MAILBOX #  *******  113  *******
    005432  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005434  000000                          HALT                    ;V NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 673

                                    10$:
    005436  103405                          BCS     11$
    005440  012737  000114  000302          MOV     #114,$FATAL     ;MOVE TO MAILBOX #  *******  114  *******
    005446  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005450  000000                          HALT                    ;C NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 665

    005452  013706  177776          11$:    MOV     CC,SP
    005456  042706  000017                  BIC     #17,SP
    005462  022706  000340                  CMP     #340,SP
    005466  001405                          BEQ     13$
    005470                          12$:
    005470  012737  000115  000302          MOV     #115,$FATAL     ;MOVE TO MAILBOX #  *******  115  *******
    005476  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005500  000000                          HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 651
    005502  012737  000022  000020  13$:    MOV     #22,20          ;.+2
    005510  005037  000022                  CLR     22              ;HALT
```

```
753                                     .SBTTL  TEST #25 - TEST THAT A TRAP OCCURS ON AN EMT  INS
                                        ;*************************************************************************
                                        ;TEST 25 - TEST THAT A TRAP OCCURS ON AN EMT  INS
                                        ;*************************************************************************
005514  032737  000001  177766  TST25:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005522  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
005524  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
005532  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005534  000000                          HALT                    ;CPU POWER BIT FOUND SET
005536  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
005544  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
005550  022737  000025  000304          CMP     #25,$TESTN      ;SEQUENCE ERROR?
005556  001006                          BNE     TST26-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
754 005560  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
755 005564  012737  005606  000030      MOV     #1$,RTRAP3      ;RETURN LOCATION
756 005572  104000                      EMT                     ;RESERVE INSTRUCTIOM, SHOULD TRAP
757 005574  012737  000116  000302      MOV     #116,$FATAL     ;MOVE TO MAILBOX #  *******  116  *******
005602  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005604  000000                          HALT                    ;EMT DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764
758 005606                          1$:
```

```
 759                                    .SBTTL  TEST #26 - TEST DEC OF STACK POINTER ON A TRAP OPER
                                        ;******************************************************************************
                                        ;TEST 26 - TEST DEC OF STACK POINTER ON A TRAP OPER
                                        ;******************************************************************************
 005606  032737  000001  177766  TST26: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 005614  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 005616  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 005624  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 005626  000000                        HALT                    ;CPU POWER BIT FOUND SET
 005630  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 005636  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 005642  022737  000026  000304        CMP     #26,$TESTN      ;SEQUENCE ERROR?
 005650  001011                        BNE     TST27-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 760 005652  012706  000500            MOV     #BUFF,SP        ;STACK POINTER SETUP
 761 005656  012737  005666  000030    MOV     #1$,RTRAP3      ;RETURN POINTER
 762 005664  104000                    EMT                     ;RESERVED INSTRUCTION
 763 005666  020627  000474    1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
 764 005672  001405                    BEQ     TST27
     005674  012737  000117  000302    MOV     #117,$FATAL     ;MOVE TO MAILBOX #  *******  117  *******
     005702  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     005704  000000                    HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

```
765                                     .SBTTL  TEST #27 - TEST THAT PROPER P.C. IS SAVED
                                        ;*************************************************************
                                        ;TEST 27 - TEST THAT PROPER P.C. IS SAVED
                                        ;*************************************************************
 005706  032737  000001  177766 TST27:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 005714  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 005716  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 005724  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 005726  000000                         HALT                    ;CPU POWER BIT FOUND SET
 005730  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 005736  005237  000304         100$:  INC     $TESTN          ;UPDATE TEST NUMBER
 005742  022737  000027  000304         CMP     #27,$TESTN      ;SEQUENCE ERROR?
 005750  001012                         BNE     TST30-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
766 005752  012706  000500             MOV     #BUFF,SP        ;STACK POINTER SETUP
767 005756  012737  005766  000030     MOV     #1$,RTRAP3      ;RTURN FROM TRAP POINTER
768 005764  104000                     EMT                     ;TRAP ON THIS INSTRICTION
769 005766  022737  005766  000474 1$:  CMP     #1$,BUFF-4      ;CHECK FOR INCREMENTED P.C.
770 005774  001405                     BEQ     TST30
 005776  012737  000120  000302         MOV     #120,$FATAL     ;MOVE TO MAILBOX #  *******  120  *******
 006004  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006006  000000                         HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 760
```

```
                                        .SBTTL  TEST #30 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 30 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
 006010  032737  000001  177766  TST30: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 006016  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 006020  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 006026  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006030  000000                        HALT                    ;CPU POWER BIT FOUND SET
 006032  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 006040  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 006044  022737  000030  000304        CMP     #30,$TESTN      ;SEQUENCE ERROR?
 006052  001040                        BNE     TST31-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 006054  012706  000500                MOV     #BUFF,SP        ;SET UP
 006060  012737  006076  000030        MOV     #1$,RTRAP3      ;SET UP
 006066  005037  177776                CLR     CC              ;CLEAR CC AND PRICRITY
 006072  000257                        CCC
 006074  104000                        EMT                     ;TRAP
 006076  023727  000476  000000  1$:   CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
 006104  001405                        BEQ     2$
 006106  012737  000121  000302        MOV     #121,$FATAL     ;MOVE TO MAILBOX #  *******  121  *******
 006114  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006116  000000                        HALT                    ;INCORRECT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
 006120  012706  000500        2$:     MOV     #BUFF,SP        ;SET UP
 006124  012737  006144  000030        MOV     #3$,RTRAP3      ;SET UP
 006132  012737  000357  177776        MOV     #357,CC         ;SET PRIORITY
 006140  000277                        SCC                     ;SET CC
 006142  104000                        EMT                     ;TRAP
 006144  023727  000476  000357  3$:   CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
 006152  001405                        BEQ     TST31
 006154  012737  000122  000302        MOV     #122,$FATAL     ;MOVE TO MAILBOX #  *******  122  *******
 006162  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006164  000000                        HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
```

```
                                        .SBTTL  TEST #31 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*************************************************************************
                                        ;TEST 31 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*************************************************************************
 006166  032737  000001  177766 TST31:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 006174  001410                 BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 006176  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 006204  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006206  000000                 HALT                    ;CPU POWER BIT FOUND SET
 006210  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 006216  005237  000304 100$:  INC     $TESTN          ;UPDATE TEST NUMBER
 006222  022737  000031  000304         CMP     #31,$TESTN      ;SEQUENCE ERROR?
 006230  001402                 BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
 006232  000137  006474                 JMP     TST32-12                ;JUMP TO ERROR HALT ON SEQ ERROR
 006236                         1000$:
 006236  012706  000500                 MOV     #BUFF,SP
 006242  012737  006256  000030         MOV     #1$,RTRAP3
 006250  005037  000032                 CLR     RTRAP3+2        ;CLEAR FUTURE PRIORITY AND CC
 006254  104000                 EMT
 006256                         1$:                             ;TEST FOR "C" CLEARED
 006256  100005                 BPL     2$
 006260  012737  000123  000302         MOV     #123,$FATAL     ;MOVE TO MAILBOX #  *******  123  *******
 006266  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006270  000000                 HALT                    ;C NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 761
 006272                         2$:
 006272  001005                 BNE     3$
 006274  012737  000124  000302         MOV     #124,$FATAL     ;MOVE TO MAILBOX #  *******  124  *******
 006302  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006304  000000                 HALT                    ;Z NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 753
 006306                         3$:
 006306  102005                 BVC     4$
 006310  012737  000125  000302         MOV     #125,$FATAL     ;MOVE TO MAILBOX #  *******  125  *******
 006316  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006320  000000                 HALT                    ;V NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 745
 006322                         4$:
 006322  103005                 BCC     5$
 006324  012737  000126  000302         MOV     #126,$FATAL     ;MOVE TO MAILBOX #  *******  126  *******
 006332  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006334  000000                 HALT                    ;C NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 737
 006336  032737  000340  177776 5$:     BIT     #340,CC         ;TEST PRIORITY
 006344  001405                 BEQ     6$
 006346  012737  000127  000302         MOV     #127,$FATAL     ;MOVE TO MAILBOX #  *******  127  *******
 006354  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006356  000000                 HALT                    ;PRIORITY NOT ZERO
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 726
 006360  012706  000500         6$:     MOV     #BUFF,SP
 006364  012737  006402  000030         MOV     #7$,RTRAP3
 006372  012737  000357  000032         MOV     #357,RTRAP3+2   ;SET NEW "CC" AND PRIORITY
```

```
    801 006400  104000                        EMT                     ;TRAP HERE
    802 006402                          7$:
        006402  100405                        BMI     8$
        006404  012737  000130  000302        MOV     #130,$FATAL     ;MOVE TO MAILBOX #  *******  130  *******
        006412  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        006414  000000                        HALT                    ;N NOT SET
                                                                       ;TO SCOPE REPLACE HALT WITH 240
                                                                       ;AND REPLACE NEXT INST WITH 707

    803 006416                          8$:
        006416  001405                        BEQ     9$
        006420  012737  000131  000302        MOV     #131,$FATAL     ;MOVE TO MAILBOX #  *******  131  *******
        006426  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        006430  000000                        HALT                    ;Z NOT SET
                                                                       ;TO SCOPE REPLACE HALT WITH 240
                                                                       ;AND REPLACE NEXT INST WITH 701

    804 006432                          9$:
        006432  102405                        BVS     10$
        006434  012737  000132  000302        MOV     #132,$FATAL     ;MOVE TO MAILBOX #  *******  132  *******
        006442  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        006444  000000                        HALT                    ;V NOT SET
                                                                       ;TO SCOPE REPLACE HALT WITH 240
                                                                       ;AND REPLACE NEXT INST WITH 673

    805 006446                          10$:
        006446  103405                        BCS     11$
        006450  012737  000133  000302        MOV     #133,$FATAL     ;MOVE TO MAILBOX #  *******  133  *******
        006456  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        006460  000000                        HALT                    ;C NOT SET
                                                                       ;TO SCOPE REPLACE HALT WITH 240
                                                                       ;AND REPLACE NEXT INST WITH 665

    806 006462  000257                  11$:  CCC
    807 006464  022737  000340  177776        CMP     #340,CC
    808 006472  001405                        BEQ     TST32
        006474  012737  000134  000302        MOV     #134,$FATAL     ;MOVE TO MAILBOX #  *******  134  *******
        006502  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        006504  000000                        HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                       ;TO SCOPE REPLACE HALT WITH 240
                                                                       ;AND REPLACE NEXT INST WITH 653
```

TEST #32 - TEST THAT ALL COMB EMT WILL CAUSE A TRAP

```
809                                             .SBTTL  TEST #32 - TEST THAT ALL COMB  EMT WILL CAUSE A TRAP
                                                ;*****************************************************************
                                                ;TEST 32 - TEST THAT ALL COMB  EMT WILL CAUSE A TRAP
                                                ;*****************************************************************
     006506  032737  000001  177766  TST32:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     006514  001410                   BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     006516  012737  000177  000302           MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     006524  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     006526  000000                   HALT                    ;CPU POWER BIT FOUND SET
     006530  042737  000001  177766           BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     006536  005237  000304         100$:    INC     $TESTN          ;UPDATE TEST NUMBER
     006542  022737  000032  000304           CMP     #32,$TESTN      ;SEQUENCE ERROR?
     006550  001011                   BNE     3$              ;BRANCH TO ERROR HALT ON SEQ ERROR
810  006552  012737  104000  006572           MOV     #EMT,2$         ;INITIALIZE BASE EMT INSTRUCTION
811  006560  012737  006606  000030           MOV     #4$,30          ;RETURN FROM TRAP TO 3$
812  006566  012706  000500         1$:      MOV     #BUFF,SP        ;SET UP STACK POINTER
813  006572  104000                 2$:      EMT                     ;TRAP INST. WILL BE MODIFIED TO EMT+377
814  006574                          3$:
     006574  012737  000135  000302           MOV     #135,$FATAL     ;MOVE TO MAILBOX #  *******  135  *******
     006602  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     006604  000000                   HALT                    ;PREVIOUS INST FAILED TO TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
815  006606  005237  006572         4$:      INC     2$              ;INCREMENT TRAP INSTRUCTION
816  006612  022737  104377  006572           CMP     #EMT+377,2$     ;EMT+377 TO EMT?
817  006620  103362                   BHIS    1$              ;HAVE WE TESTED ALL
818                                                             ;YES
819  006622  012737  000032  000030           MOV     #32,30          ;/.+
820  006630  005037  000032           CLR     32              ;HALT
```

```
821                                     .SBTTL  TEST #33 - TEST THAT A TRAP OCCURES ON AN "TRACE-TRT" INS
                                        ;*************************************************************************
                                        ;TEST 33 - TEST THAT A TRAP OCCURES ON AN "TRACE-TRT" INS
                                        ;*************************************************************************
006634  032737  000001  177766  TST33:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006642  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
006644  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
006652  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006654  000000                          HALT                    ;CPU POWER BIT FOUND SET
006656  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
006664  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
006670  022737  000033  000304          CMP     #33,$TESTN      ;SEQUENCE ERROR?
006676  001006                          BNE     TST34-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
822 006700  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
823 006704  012737  006726  000014      MOV     #1$,RTRAP4      ;RETURN LOCATION
824 006712  000003                      TRT                     ;RESERVED INSTRUCTION, SHOULD TRAP
825 006714  012737  000136  000302      MOV     #136,$FATAL     ;MOVE TO MAILBOX #  *******  136  *******
006722  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006724  000000                          HALT                    ;TRT DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764

826 006726                          1$:
```

```
827                                     .SBTTL  TEST #34 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;******************************************************************************
                                        ;
                                        ;TEST 34 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;******************************************************************************
     006726  032737  000001  177766  TST34:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     006734  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     006736  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     006744  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     006746  000000                          HALT                    ;CPU POWER BIT FOUND SET
     006750  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     006756  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     006762  022737  000034  000304          CMP     #34,$TESTN      ;SEQUENCE ERROR?
     006770  001011                          BNE     TST35-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
828  006772  012706  000500                  MOV     #BUFF,SP        ;STACK POINTER SETUP
829  006776  012737  007006  000014          MOV     #1$,RTRAP4      ;RETURN POINTER
830  007004  000003                          TRT                     ;RESERVED INSTRUCTION
831  007006  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
832  007012  001405                          BEQ     TST35
     007014  012737  000137  000302          MOV     #137,$FATAL     ;MOVE TO MAILBOX #  *******  137  *******
     007022  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     007024  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 761
```

TEST #35 - TEST THAT PROPER P.C. IS SAVED

```
                                        .SBTTL  TEST #35 - TEST THAT PROPER P.C. IS SAVED
                                        ;*************************************************************************
                                        ;TEST 35 - TEST THAT PROPER P.C. IS SAVED
                                        ;*************************************************************************
 007026  032737  000001  177766  TST35: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 007034  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 007036  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 007044  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 007046  000000                        HALT                    ;CPU POWER BIT FOUND SET
 007050  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 007056  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 007062  022737  000035  000304        CMP     #35,$TESTN      ;SEQUENCE ERROR?
 007070  001012                        BNE     TST36-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 834 007072  012706  000500            MOV     #BUFF,SP        ;STACK POINTER SETUP
 835 007076  012737  007106  000014    MOV     #1$,RTRAP4      ;RETURN FROM TRAP POINTER
 836 007104  000003                    TRT                     ;TRAP ON THIS INSTRUCTION
 837 007106  022737  007106  000474  1$:     CMP     #.,BUFF-4       ;CHECK FOR INCREMENTED P.C.
 838 007114  001405                    BEQ     TST36
     007116  012737  000140  000302        MOV     #140,$FATAL     ;MOVE TO MAILBOX #  *******  140  *******
     007124  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     007126  000000                        HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 760
```

TEST #36 - TEST THAT 'OLD" CC AND PRI ARE PLACED ON STACK

```
839                                     .SBTTL  TEST #36 - TEST THAT 'OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 36 - TEST THAT 'OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
     007130  032737  000001  177766  TST36:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     007136  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     007140  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     007146  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     007150  000000                          HALT                    ;CPU POWER BIT FOUND SET
     007152  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     007160  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     007164  022737  000036  000304          CMP     #36,$TESTN      ;SEQUENCE ERROR?
     007172  001040                          BNE     TST37-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
840  007174  012706  000500                  MOV     #BUFF,SP        ;SET UP
841  007200  012737  007216  000014          MOV     #1$,RTRAP4      ;SET UP
842  007206  005037  177776                  CLR     CC              ;CLEAR CC AND PRICRITY
843  007212  000257                          CCC
844  007214  000003                          TRT                     ;TRAP
845  007216  023727  000476  000000  1$:     CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
846                                                                  ;TEST FOR ALL ZEROS
847  007224  001405                          BEQ     2$
     007226  012737  000141  000302          MOV     #141,$FATAL     ;MOVE TO MAILBOX #  *******  141  *******
     007234  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     007236  000000                          HALT                    ;INCORRECT STATUS
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 755
848  007240  012706  000500          2$:     MOV     #BUFF,SP        ;SET UP
849  007244  012737  007264  000014          MOV     #3$,RTRAP4      ;SET UP
850  007252  012737  000357  177776          MOV     #357,CC         ;SET PRIORITY
851  007260  000277                          SCC                     ;SET-SET CC
852  007262  000003                          TRT                     ;TRAP
853  007264  023727  000476  000357  3$:     CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
854  007272  001405                          BEQ     TST37
     007274  012737  000142  000302          MOV     #142,$FATAL     ;MOVE TO MAILBOX #  *******  142  *******
     007302  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     007304  000000                          HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 732
```

TEST #37 - TEST THAT 'NEW' STATUS IS CORRECT

```
855                                     .SBTTL  TEST #37 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*****************************************************************
                                        ;TEST 37 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*****************************************************************
007306  032737  000001  177766  TST37:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007314  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
007316  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
007324  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007326  000000                          HALT                    ;CPU POWER BIT FOUND SET
007330  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
007336  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
007342  022737  000037  000304          CMP     #37,$TESTN      ;SEQUENCE ERROR?
007350  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
007352  000137  007620                  JMP     15$             ;JUMP TO ERROR HALT ON SEQ ERROR
007356                          1000$:
856 007356  012706  000500              MOV     #BUFF,SP
857 007362  012737  007376  000014      MOV     #4$,RTRAP4
858 007370  005037  000016              CLR     RTRAP4+2        ;CLEAR FUTURE PRIORITY AND CC
859 007374  000003                      TPT
860 007376                      4$:                             ;TEST FOR "C" CLEARED
861 007376  100005                      BPL     5$
007400  012737  000143  000302          MOV     #143,$FATAL     ;MOVE TO MAILBOX #  *******  143  *******
007406  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007410  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
862 007412                      5$:
007412  001005                      BNE     6$
007414  012737  000144  000302          MOV     #144,$FATAL     ;MOVE TO MAILBOX #  *******  144  *******
007422  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007424  000000                          HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
863 007426                      6$:
007426  102005                      BVC     7$
007430  012737  000145  000302          MOV     #145,$FATAL     ;MOVE TO MAILBOX #  *******  145  *******
007436  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007440  000000                          HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
864 007442                      7$:
007442  103005                      BCC     8$
007444  012737  000146  000302          MOV     #146,$FATAL     ;MOVE TO MAILBOX #  *******  146  *******
007452  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007454  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
865 007456  032737  000340  177776  8$:  BIT     #340,CC         ;TEST PRIORITY
866 007464  001405                      BEQ     9$
007466  012737  000147  000302          MOV     #147,$FATAL     ;MOVE TO MAILBOX #  *******  147  *******
007474  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007476  000000                          HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 726
867 007500  012706  000500          9$:  MOV     #BUFF,SP
868 007504  012737  007522  000014      MOV     #10$,RTRAP4
869 007512  012737  000357  000016      MOV     #357,RTRAP4+2   ;SET NEW "CC" AND PRIORITY
```

TEST #37 - TEST THAT 'NEW' STATUS IS CORRECT

```
870 007520  000003                        TRT                     ;TRAP HERE
871 007522                          10$:
    007522  100405                        BMI     11$
    007524  012737  000150  000302        MOV     #150,$FATAL     ;MOVE TO MAILBOX #  *******  150  *******
    007532  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007534  000000                        HALT                    ;N NOT SET
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 707
872 007536                          11$:
    007536  001405                        BEQ     12$
    007540  012737  000151  000302        MOV     #151,$FATAL     ;MOVE TO MAILBOX #  *******  151  *******
    007546  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007550  000000                        HALT                    ;Z NOT SET
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 701
873 007552                          12$:
    007552  102405                        BVS     13$
    007554  012737  000152  000302        MOV     #152,$FATAL     ;MOVE TO MAILBOX #  *******  152  *******
    007562  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007564  000000                        HALT                    ;V NOT SET
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 673
874 007566                          13$:
    007566  103405                        BCS     14$
    007570  012737  000153  000302        MOV     #153,$FATAL     ;MOVE TO MAILBOX #  *******  153  *******
    007576  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007600  000000                        HALT                    ;C NOT SET
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 665
875 007602  013706  177776          14$:  MOV     CC,SP
876 007606  042706  000017                BIC     #17,SP
877 007612  022706  000340                CMP     #340,SP
878 007616  001405                        BEQ     16$
    007620                          15$:
    007620  012737  000154  000302        MOV     #154,$FATAL     ;MOVE TO MAILBOX #  *******  154  *******
    007626  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007630  000000                        HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 651
879 007632  012737  000016  000014  16$:  MOV     #16,14
880 007640  005037  000016                CLR     16
881
882                                 ;PDP-11 ILLEGAL AND ADDRESS INSTRUCTION TEST
883                                 ;ALL INSTRUCTIONS THAT ARE ILLEGAL
884                                 ;SHOULD TRAP TO LOCATION 10, AND THE
885                                 ;PC THAT POINTS TO THE TRAPPING INSTRUCTION
886                                 ;SHOULD BE PLACED ON THE STACK
887
```

```
888                                     .SBTTL  TEST #40 - TEST THAT A TRAP OCCURS ON AN ILLEGAL INS
                                        ;*****************************************************************
                                        ;TEST 40 - TEST THAT A TRAP OCCURS ON AN ILLEGAL INS
                                        ;*****************************************************************
007644  032737  000001  177766  TST40:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007652  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
007654  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
007662  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007664  000000                          HALT                    ;CPU POWER BIT FOUND SET
007666  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
007674  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
007700  022737  000040  000304          CMP     #40,$TESTN      ;SEQUENCE ERROR?
007706  001006                          BNE     TST41-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
889 007710  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
890 007714  012737  007736  000010      MOV     #1$,RTRAP       ;RETURN LOCATION
891 007722  000100                      JMP     R0              ;ILLEGAL INSTRUCTION, SHOULD TRAP
892 007724  012737  000155  000302      MOV     #155,$FATAL     ;MOVE TO MAILBOX #  *******  155  *******
    007732  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007734  000000                      HALT                    ;ILLEGAL INSTRUCTION DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764

893 007736                      1$:
```

```
894                                     .SBTTL  TEST #41 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*************************************************************************
                                        ;TEST 41 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*************************************************************************
007736  032737  000001  177766  TST41:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007744  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
007746  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
007754  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007756  000000                          HALT                    ;CPU POWER BIT FOUND SET
007760  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
007766  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
007772  022737  000041  000304          CMP     #41,$TESTN      ;SEQUENCE ERROR?
010000  001011                          BNE     TST42-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
895 010002  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
896 010006  012737  010016  000010      MOV     #1$,RTRAP       ;RETURN POINTER
897 010014  000100                      JMP     R0              ;RESERVED INSTRUCTION
898 010016  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
899 010022  001405                      BEQ     TST42
010024  012737  000156  000302          MOV     #156,$FATAL     ;MOVE TO MAILBOX #  *******  156  *******
010032  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010034  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

TEST #42 - TEST THAT PROPER P.C. IS SAVED

```
900                                     .SBTTL  TEST #42 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
                                        ;TEST 42 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
010036  032737  000001  177766  TST42:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
010044  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
010046  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
010054  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010056  000000                          HALT                    ;CPU POWER BIT FOUND SET
010060  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
010066  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
010072  022737  000042  000304          CMP     #42,$TESTN      ;SEQUENCE ERROR?
010100  001012                          BNE     TST43-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
901 010102  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
902 010106  012737  010116  000010      MOV     #1$,RTRAP       ;RETURN FROM TRAP POINTER
903 010114  000100                      JMP     R0              ;TRAP ON THIS INSTRUCTION
904 010116  022737  010116  000474  1$:  CMP     #.,BUFF-4       ;CHECK FOR INCREMENTED P.C.
905 010124  001405                      BEQ     TST43
    010126  012737  000157  000302      MOV     #157,$FATAL     ;MOVE TO MAILBOX #  *******  157  *******
    010134  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010136  000000                      HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
```

```
906                                             .SBTTL  TEST #43 - TEST THAT 'OLD'' CC AND PRI ARE PLACED ON STACK
                                                ;*****************************************************************
                                                ;TEST 43 - TEST THAT 'OLD'' CC AND PRI ARE PLACED ON STACK
                                                ;*****************************************************************
    010140  032737  000001  177766  TST43:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
    010146  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
    010150  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
    010156  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010160  000000                          HALT                    ;CPU POWER BIT FOUND SET
    010162  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
    010170  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
    010174  022737  000043  000304          CMP     #43,$TESTN      ;SEQUENCE ERROR?
    010202  001040                          BNE     TST44-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
907 010204  012706  000500                  MOV     #BUFF,SP        ;SET UP
908 010210  012737  010226  000010          MOV     #1$,RTRAP       ;SET UP
909 010216  005037  177776                  CLR     CC              ;CLEAR CC AND PRICRITY
910 010222  000257                          CCC
911 010224  000100                          JMP     R0              ;TRAP
912 010226  023727  000476  000000  1$:     CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
913 010234  001405                          BEQ     2$
    010236  012737  000160  000302          MOV     #160,$FATAL     ;MOVE TO MAILBOX #  *******  160  *******
    010244  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010246  000000                          HALT                    ;INCORRECT STATUS
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 755
914 010250  012706  000500          2$:     MOV     #BUFF,SP        ;SET UP
915 010254  012737  010274  000010          MOV     #3$,RTRAP       ;SET UP
916 010262  012737  000357  177776          MOV     #357,CC         ;SET PRIORITY
917 010270  000277                          SCC                     ;SET CC
918 010272  000100                          JMP     R0              ;TRAP
919 010274  023727  000476  000357  3$:     CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
920 010302  001405                          BEQ     TST44
    010304  012737  000161  000302          MOV     #161,$FATAL     ;MOVE TO MAILBOX #  *******  161  *******
    010312  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010314  000000                          HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 732
```

```
                                        .SBTTL  TEST #44 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*************************************************************
                                        ;TEST 44 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*************************************************************
 010316  032737  000001  177766 TST44:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 010324  001410                 BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 010326  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 010334  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010336  000000                 HALT                    ;CPU POWER BIT FOUND SET
 010340  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 010346  005237  000304 100$:  INC     $TESTN          ;UPDATE TEST NUMBER
 010352  022737  000044  000304         CMP     #44,$TESTN      ;SEQUENCE ERROR?
 010360  001402                 BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
 010362  000137  010624                 JMP     TST45-12                ;JUMP TO ERROR HALT ON SEQ ERROR
 010366                         1000$:
 010366  012706  000500                 MOV     #BUFF,SP
 010372  012737  010406  000010         MOV     #1$,RTRAP
 010400  005037  000012                 CLR     RTRAP+2 ;CLEAR FUTURE PRIORITY AND CC
 010404  000100                 JMP     R0
 010406                         1$:                     ;TEST FOR "C" CLEARED
 010406  100005                 BPL     2$
 010410  012737  000162  000302         MOV     #162,$FATAL     ;MOVE TO MAILBOX #  *******  162  *******
 010416  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010420  000000                 HALT                    ;C NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 761
 010422                         2$:
 010422  001005                 BNE     3$
 010424  012737  000163  000302         MOV     #163,$FATAL     ;MOVE TO MAILBOX #  *******  163  *******
 010432  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010434  000000                 HALT                    ;Z NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 753
 010436                         3$:
 010436  102005                 BVC     4$
 010440  012737  000164  000302         MOV     #164,$FATAL     ;MOVE TO MAILBOX #  *******  164  *******
 010446  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010450  000000                 HALT                    ;V NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 745
 010452                         4$:
 010452  103005                 BCC     5$
 010454  012737  000165  000302         MOV     #165,$FATAL     ;MOVE TO MAILBOX #  *******  165  *******
 010462  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010464  000000                 HALT                    ;C NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 737
 010466  032737  000357  177776 5$:     BIT     #357,CC         ;TEST PRIORITY
 010474  001405                 BEQ     6$
 010476  012737  000166  000302         MOV     #166,$FATAL     ;MOVE TO MAILBOX #  *******  166  *******
 010504  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010506  000000                 HALT                    ;PRIORITY NOT ZERO
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 726
 010510  012706  000500         6$:     MOV     #BUFF,SP
 010514  012737  010532  000010         MOV     #7$,RTRAP
 010522  012737  000357  000012         MOV     #357,RTRAP+2    ;SET NEW "CC" AND PRIORITY
```

```
936 010530  000100                          JMP     R0              ;TRAP HERE
937 010532                          7$:
    010532  100405                          BMI     8$
    010534  012737  000167  000302          MOV     #167,$FATAL     ;MOVE TO MAILBOX #  *******  167  *******
    010542  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010544  000000                          HALT                    ;N NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 707
938 010546                          8$:
    010546  001405                          BEQ     9$
    010550  012737  000170  000302          MOV     #170,$FATAL     ;MOVE TO MAILBOX #  *******  170  *******
    010556  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010560  000000                          HALT                    ;Z NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 701
939 010562                          9$:
    010562  102405                          BVS     10$
    010564  012737  000171  000302          MOV     #171,$FATAL     ;MOVE TO MAILBOX #  *******  171  *******
    010572  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010574  000000                          HALT                    ;V NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 673
940 010576                          10$:
    010576  103405                          BCS     11$
    010600  012737  000172  000302          MOV     #172,$FATAL     ;MOVE TO MAILBOX #  *******  172  *******
    010606  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010610  000000                          HALT                    ;C NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 665
941 010612  013706  177776          11$:    MOV     CC,SP
942 010616  022706  000357                  CMP     #357,SP
943 010622  001405                          BEQ     TST45
    010624  012737  000173  000302          MOV     #173,$FATAL     ;MOVE TO MAILBOX #  *******  173  *******
    010632  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010634  000000                          HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 653
```

TEST #45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS

```
944                                     .SBTTL  TEST #45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS
                                        ;*****************************************************************
                                        ;TEST 45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS
                                        ;*****************************************************************
010636  032737  000001  177766  TST45:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
010644  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
010646  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
010654  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010656  000000                          HALT                    ;CPU POWER BIT FOUND SET
010660  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
010666  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
010672  022737  000045  000304          CMP     #45,$TESTN      ;SEQUENCE ERROR?
010700  001006                          BNE     TST46-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
945 010702  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
946 010706  012737  010730  000010      MOV     #1$,RTRAP       ;RETURN LOCATION
947 010714  004000                      JSR     R0,R0           ;RESERVED INS, SHOULD TRAP
948 010716  012737  000174  000302      MOV     #174,$FATAL     ;MOVE TO MAILBOX #  *******  174  *******
010724  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010726  000000                          HALT                    ;DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764

949 010730                      1$:
```

```
    950                                     .SBTTL  TEST #46 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                            ;*****************************************************************************
                                            ;TEST 46 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                            ;*****************************************************************************
    010730  032737  000001  177766  TST46:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
    010736  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
    010740  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
    010746  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010750  000000                          HALT                    ;CPU POWER BIT FOUND SET
    010752  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
    010760  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
    010764  022737  000046  000304          CMP     #46,$TESTN      ;SEQUENCE ERROR?
    010772  001011                          BNE     TST47-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
    951 010774  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
    952 011000  012737  0110 0  000010      MOV     #1$,RTRAP       ;RETURN POINTER
    953 011006  004000                      JSR     R0,R0           ;RESERVED INS
    954 011010  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
    955 011014  001405                          BEQ     TST47
        011016  012737  000175  000302          MOV     #175,$FATAL     ;MOVE TO MAILBOX #  *******  175  *******
        011024  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        011026  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 761
```

TEST #47 - TEST THAT PROPER P.C. IS SAVED

```
956                                     .SBTTL  TEST #47 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************************
                                        ;TEST 47 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************************
011030  032737  000001  177766  TST47:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011036  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
011040  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
011046  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011050  000000                          HALT                    ;CPU POWER BIT FOUND SET
011052  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
011060  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
011064  022737  000047  000304          CMP     #47,$TESTN      ;SEQUENCE ERROR?
011072  001012                          BNE     TST50-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
957 011074  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
958 011100  012737  011110  000010      MOV     #1$,RTRAP       ;RETURN FROM TRAP POINTER
959 011106  004000                      JSR     R0,R0           ;TRAP ON THIS INS
960 011110  022737  011110  000474  1$:  CMP     #1$,BUFF-4      ;CHECK FOR INCREMENED P.C.
961 011116  001405                      BEQ     TST50
    011120  012737  000176  000302      MOV     #176,$FATAL     ;MOVE TO MAILBOX #  *******  176  *******
    011126  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011130  000000                      HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
962
```

```
963                                     .SBTTL  TEST #50 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                        ;*************************************************************
                                        ;TEST 50 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                        ;*************************************************************
    011132  032737  000001  177766  TST50:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
    011140  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
    011142  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
    011150  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011152  000000                          HALT                    ;CPU POWER BIT FOUND SET
    011154  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
    011162  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
    011166  022737  000050  000304          CMP     #50,$TESTN      ;SEQUENCE ERROR?
    011174  001040                          BNE     TST51-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
964 011176  012706  000500                  MOV     #BUFF,SP        ;SET UP
965 011202  012737  011220  000010          MOV     #1$,RTRAP       ;SET UP
966 011210  005037  177776                  CLR     CC              ;CLEAR CC AND PRICRITY
967 011214  000257                          CCC
968 011216  004000                          JSR     R0,R0           ;TRAP
969 011220  023727  000476  000000  1$:     CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
970 011226  001405                          BEQ     2$
    011230  012737  000177  000302          MOV     #177,$FATAL     ;MOVE TO MAILBOX #  *******  177  *******
    011236  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011240  000000                          HALT                    ;INCORRCT STATUS
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 755
971 011242  012706  000500          2$:     MOV     #BUFF,SP        ;SET UP
972 011246  012737  011266  000010          MOV     #3$,RTRAP       ;SET UP
973 011254  012737  000357  177776          MOV     #357,CC         ;SET PRIORITY
974 011262  000277                          SCC                     ;SET CC
975 011264  004000                          JSR     R0,R0           ;TRAP
976 011266  023727  000476  000357  3$:     CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
977 011274  001405                          BEQ     TST51
    011276  012737  000200  000302          MOV     #200,$FATAL     ;MOVE TO MAILBOX #  *******  200  *******
    011304  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011306  000000                          HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 732
```

TEST #51 - TEST THAT 'NEW' STATUS IS CORRECT

```
                                        .SBTTL  TEST #51 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*****************************************************************
                                        ;TEST 51 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*****************************************************************
011310  032737  000001  177766 TST51:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011316  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
011320  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
011326  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011330  000000                          HALT                    ;CPU POWER BIT FOUND SET
011332  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
011340  005237  000304        100$:     INC     $TESTN          ;UPDATE TEST NUMBER
011344  022737  000051  000304          CMP     #51,$TESTN      ;SEQUENCE ERROR?
011352  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
011354  000137  011614                  JMP     TST52-12                ;JUMP TO ERROR HALT ON SEQ ERROR
011360                          1000$:
979 011360  012706  000500              MOV     #BUFF,SP
980 011364  012737  011400  000010      MOV     #1$,RTRAP
981 011372  005037  000012              CLR     RTRAP+2         ;CLEAR FUTURE PRIORITY AND CC
982 011376  004000                      JSR     R0,R0
983 011400                      1$:             ;TEST FOR "C" CLEARED
984 011400  100005                      BPL     2$
    011402  012737  000201  000302      MOV     #201,$FATAL     ;MOVE TO MAILBOX #  *******  201  *******
    011410  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011412  000000                      HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
985 011414                      2$:
    011414  001005                      BNE     3$
    011416  012737  000202  000302      MOV     #202,$FATAL     ;MOVE TO MAILBOX #  *******  202  *******
    011424  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011426  000000                      HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
986 011430                      3$:
    011430  102005                      BVC     4$
    011432  012737  000203  000302      MOV     #203,$FATAL     ;MOVE TO MAILBOX #  *******  203  *******
    011440  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011442  000000                      HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
987 011444                      4$:
    011444  103005                      BCC     5$
    011446  012737  000204  000302      MOV     #204,$FATAL     ;MOVE TO MAILBOX #  *******  204  *******
    011454  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011456  000000                      HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
988 011460  013700  177776      5$:     MOV     CC,R0           ;TEMP STORAGE
989 011464  001405                      BEQ     6$
    011466  012737  000205  000302      MOV     #205,$FATAL     ;MOVE TO MAILBOX #  *******  205  *******
    011474  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011476  000000                      HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 727
990 011500  012706  000500      6$:     MOV     #BUFF,SP
991 011504  012737  011522  000010      MOV     #7$,RTRAP
992 011512  012737  000357  000012      MOV     #357,RTRAP+2    ;SET NEW "CC" AND PRIORITY
```

TEST #51 - TEST THAT "NEW" STATUS IS CORRECT

```
                                                JSR     R0,R0           ;TRAP HERE
                                        7$:
011522  100405                                  BMI     8$
011524  012737  000206  000302                  MOV     #206,$FATAL     ;MOVE TO MAILBOX #  *******  206  *******
011532  005212                                  INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011534  000000                                  HALT                    ;N NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 710

                                        8$:
011536  001405                                  BEQ     9$
011540  012737  000207  000302                  MOV     #207,$FATAL     ;MOVE TO MAILBOX #  *******  207  *******
011546  005212                                  INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011550  000000                                  HALT                    ;Z NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 702

                                        9$:
011552  102405                                  BVS     10$
011554  012737  000210  000302                  MOV     #210,$FATAL     ;MOVE TO MAILBOX #  *******  210  *******
011562  005212                                  INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011564  000000                                  HALT                    ;V NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 674

                                        10$:
011566  103405                                  BCS     11$
011570  012737  000211  000302                  MOV     #211,$FATAL     ;MOVE TO MAILBOX #  *******  211  *******
011576  005212                                  INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011600  000000                                  HALT                    ;C NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 666

011602  013700  177776          11$:    MOV     CC,R0
011606  022700  000357                  CMP     #357,R0
011612  001405                          BEQ     TST52
011614  012737  000212  000302          MOV     #212,$FATAL     ;MOVE TO MAILBOX #  *******  212  *******
011622  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011624  000000                          HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 654
```

```
                                        .SBTTL  TEST #52 - TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS
                                        ;*************************************************************************
                                        ;TEST 52 - TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS
                                        ;*************************************************************************
011626  032737  000001  177766  TST52:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011634  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
011636  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
011644  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011646  000000                         HALT                    ;CPU POWER BIT FOUND SET
011650  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
011656  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
011662  022737  000052  000304         CMP     #52,$TESTN      ;SEQUENCE ERROR?
011670  001040                         BNE     3$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1003 011672  005037  177766           CLR     CPUERR          ;CLEAR CPU ERROR REGISTER
1004 011676  005737  177766           TST     CPUERR          ;VERIFY THAT IT CLEARED
1005 011702  001405                   BEQ     1$
011704  012737  000213  000302         MOV     #213,$FATAL     ;MOVE TO MAILBOX #  *******  213  *******
011712  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011714  000000                         HALT                    ;CPU ERROR REG FAILED TO CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 765
1006 011716  012706  000500   1$:      MOV     #BUFF,SP        ;STACK POINTER SETUP
1007 011722  012737  011746  000004   MOV     #2$,RTRAP5      ;RETURN LOCATION
1008 011730  005737  000001           TST     1               ;ILLEGAL ADDRESS INS. SHOULD TRAP
1009 011734  012737  000214  000302   MOV     #214,$FATAL     ;MOVE TO MAILBOX #  *******  214  *******
011742  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011744  000000                         HALT                    ;ILLEGAL ADDRESS DID NOT TRAP
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 751
1010 011746                    2$:
1011 011746  013737  177766  000502   MOV     CPUERR,RCPUER   ;READ AND SAVE CPU ERROR REGISTER
1012 011754  042737  177413  000502   BIC     #CERMSK,RCPUER  ;MASK OFF UNUSED BITS OF CPU ERROR REG
1013 011762  022737  000100  000502   CMP     #100,RCPUER     ;ODD ADDRESS BIT SET?
1014 011770  001405                   BEQ     4$
011772                         3$:
011772  012737  000215  000302         MOV     #215,$FATAL     ;MOVE TO MAILBOX #  *******  215  *******
012000  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
012002  000000                         HALT                    ;INCORRECT CPU ERROR REG CONTENTS, OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
1015 012004  005037  177766   4$:      CLR     CPUERR          ;CLEAR ODD ADDRESS BIT
1016
```

```
1017                                    .SBTTL  TEST #53 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************************
                                        ;TEST 53 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************************
012010  032737  000001  177766  TST53:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
012016  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
012020  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
012026  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
012030  000000                          HALT                    ;CPU POWER BIT FOUND SET
012032  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
012040  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
012044  022737  000053  000304          CMP     #53,$TESTN      ;SEQUENCE ERROR?
012052  001012                          BNE     TST54-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1018 012054  012706  000500            MOV     #BUFF,SP        ;STACK POINTER SETUP
1019 012060  012737  012072  000004    MOV     #1$,RTRAPS      ;RETURN POINTER
1020 012066  005737  000001            TST     1               ;RESERVED INS
1021 012072  020627  000474    1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
1022 012076  001405                    BEQ     TST54
012100  012737  000216  000302          MOV     #216,$FATAL     ;MOVE TO MAILBOX #  *******  216  *******
012106  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
012110  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
```

```
1023                                    .SBTTL  TEST #54 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
                                        ;TEST 54 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
012112  032737  000001  177766  TST54:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
012120  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
012122  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
012130  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
012132  000000                          HALT                    ;CPU POWER BIT FOUND SET
012134  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
012142  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
012146  022737  000054  000304          CMP     #54,$TESTN      ;SEQUENCE ERROR?
012154  001013                          BNE     TST55-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1024 012156  012706  000500            MOV     #BUFF,SP        ;STACK POINTER SETUP
1025 012162  012737  012174  000004    MOV     #1$,RTRAP5      ;RETURN FROM TRAP POINTER
1026 012170  005737  000001            TST     1               ;TRAP ON THIS INSTRUCTION
1027 012174  022737  012174  000474  1$:     CMP     #1$,BUFF-4      ;CHECK FOR INCREMENTED P.C.
1028 012202  001405                          BEQ     TST55
012204  012737  000217  000302          MOV     #217,$FATAL     ;MOVE TO MAILBOX #  *******  217  *******
012212  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
012214  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 757
```

```
1029                                    .SBTTL  TEST #55 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 55 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
     012216  032737  000001  177766  TST55:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     012224  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     012226  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     012234  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012236  000000                          HALT                    ;CPU POWER BIT FOUND SET
     012240  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     012246  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     012252  022737  000055  000304          CMP     #55,$TESTN      ;SEQUENCE ERROR?
     012260  001050                          BNE     TST56-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1030 012262  000405                          BR      90$             ;BRANCH OVER UNUSED ERROR CODE
1031 012264  012737  000220  000302          MOV     #220,$FATAL     ;MOVE TO MAILBOX #  *******  220  *******
     012272  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012274  000000                          HALT                    ;ERROR 177 RESERVED FOR CPU POWER BIT FOUND SET
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 771
1032 012276  012706  000500          90$:    MOV     #BUFF,SP        ;SET UP
1033 012302  012737  012322  000004          MOV     #1$,RTRAP5      ;SET UP
1034 012310  005037  177776                  CLR     CC              ;CLEAR CC AND PRIORITY
1035 012314  000257                          CCC
1036 012316  005737  000001                  TST     1               ;TRAP
1037 012322  023727  000476  000000  1$:     CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
1038 012330  001405                          BEQ     2$
     012332  012737  000221  000302          MOV     #221,$FATAL     ;MOVE TO MAILBOX #  *******  221  *******
     012340  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012342  000000                          HALT                    ;INCORRECT STATUS
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 746
1039 012344  012706  000500          2$:     MOV     #BUFF,SP        ;SET UP
1040 012350  012737  012372  000004          MOV     #3$,RTRAP5      ;SET UP
1041 012356  012737  000357  177776          MOV     #357,CC         ;SET PRIORITY
1042 012364  000277                          SCC                     ;SET CC
1043 012366  005737  000001                  TST     1               ;TRAP
1044 012372  023727  000476  000357  3$:     CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
1045 012400  001405                          BEQ     TST56
     012402  012737  000222  000302          MOV     #222,$FATAL     ;MOVE TO MAILBOX #  *******  222  *******
     012410  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012412  000000                          HALT                    ;INCORRCT STATUS ON STACK,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 722
```

```
1046                                    .SBTTL  TEST #56 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;******************************************************************
                                        ;TEST 56 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;******************************************************************
     012414  032737  000001  177766  TST56:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     012422  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     012424  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     012432  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012434  000000                          HALT                    ;CPU POWER BIT FOUND SET
     012436  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     012444  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     012450  022737  000056  000304          CMP     #56,$TESTN      ;SEQUENCE ERROR?
     012456  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
     012460  000137  012726                  JMP     TST57-12                ;JUMP TO ERROR HALT ON SEQ ERROR
     012464                          1000$:
1047 012464  012706  000500                  MOV     #BUFF,SP
1048 012470  012737  012506  000004          MOV     #1$,RTRAP5
1049 012476  005037  000006                  CLR     RTRAP5+2        ;CLEAR FUTURE PRIORITY AND CC
1050 012502  005737  000001                  TST     1               ;TRAP HERE
1051 012506                          1$:                             ;TEST FOR "C" CLEARED
1052 012506  100005                          BPL     2$
     012510  012737  000223  000302          MOV     #223,$FATAL     ;MOVE TO MAILBOX #  *******  223  *******
     012516  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012520  000000                          HALT                    ;C NOT CLEARED
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 760
1053 012522                          2$:
     012522  001005                          BNE     3$
     012524  012737  000224  000302          MOV     #224,$FATAL     ;MOVE TO MAILBOX #  *******  224  *******
     012532  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012534  000000                          HALT                    ;Z NOT CLEARED
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 752
1054 012536                          3$.
     012536  102005                  BVC     4$
     012540  012737  000225  000302          MOV     #225,$FATAL     ;MOVE TO MAILBOX #  *******  225  *******
     012546  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012550  000000                          HALT                    ;V NOT CLEARED
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 744
1055 012552                          4$:
     012552  103005                  BCC     5$
     012554  012737  000226  000302          MOV     #226,$FATAL     ;MOVE TO MAILBOX #  *******  226  *******
     012562  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012564  000000                          HALT                    ;C NOT CLEARED
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 736
1056 012566  032737  000357  177776  5$:     BIT     #357,CC         ;TEST PRIORITY FOR ZERO
1057 012574  001405                          BEQ     6$
     012576  012737  000227  000302          MOV     #227,$FATAL     ;MOVE TO MAILBOX #  *******  227  *******
     012604  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012606  000000                          HALT                    ;PRIORITY NOT ZERO
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 725
1058 012610  012706  000500          6$:     MOV     #BUFF,SP
1059 012614  012737  012634  000004          MOV     #7$,RTRAP5
1060 012622  012737  000357  000006          MOV     #357,RTRAP5+2   ;SET NEW "CC" AND PRIORITY
```

```
1061 012630  005737  000001                    TST     1               ;TRACE HERE
1062 012634                              7$:
     012634  100405                             BMI     8$
     012636  012737  000230  000302             MOV     #230,$FATAL     ;MOVE TO MAILBOX #  *******  230  *******
     012644  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012646  000000                             HALT                    ;N NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 705

1063 012650                              8$:
     012650  001405                             BEQ     9$
     012652  012737  000231  000302             MOV     #231,$FATAL     ;MOVE TO MAILBOX #  *******  231  *******
     012660  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012662  000000                             HALT                    ;Z NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 677

1064 012664                              9$:
     012664  102405                             BVS     10$
     012666  012737  000232  000302             MOV     #232,$FATAL     ;MOVE TO MAILBOX #  *******  232  *******
     012674  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012676  000000                             HALT                    ;V NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 671

1065 012700                              10$:
     012700  103405                             BCS     11$
     012702  012737  000233  000302             MOV     #233,$FATAL     ;MOVE TO MAILBOX #  *******  233  *******
     012710  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012712  000000                             HALT                    ;C NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 663

1066 012714  013700  177776              11$:   MOV     CC,R0
1067 012720  022700  000357                     CMP     #357,R0
1068 012724  001405                             BEQ     TST57
     012726  012737  000234  000302             MOV     #234,$FATAL     ;MOVE TO MAILBOX #  *******  234  *******
     012734  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012736  000000                             HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 651
```

```
:069                                    .SBTTL  TEST #57 - TEST THAT DEC R6 TO A VALUE LESS  400 TRAPS
                                        ;*************************************************************************
                                        ;TEST 57 - TEST THAT DEC R6 TO A VALUE LESS  400 TRAPS
                                        ;*************************************************************************
     012740  032737  000001  177766  TST57:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     012746  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     012750  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     012756  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012760  000000                          HALT                    ;CPU POWER BIT FOUND SET
     012762  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     012770  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     012774  022737  000057  000304          CMP     #57,$TESTN      ;SEQUENCE ERROR?
     013002  001027                          BNE     2$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1070 013004  005037  177766                  CLR     CPUERR          ;CLEAR CPU ERROR REGISTER
1071 013010  012706  000150                  MOV     #150,R6         ;R6 = 150
1072 013014  012737  013036  000004          MOV     #1$,4           ;STACK OVERFLOW TRAP POINTER
1073 013022  005746                          TST     -(R6)           ;WITH R6 = 150 SHOULD TRAP
1074 013024  012737  000235  000302          MOV     #235,$FATAL     ;MOVE TO MAILBOX #  *******  235  *******
     013032  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     013034  000000                          HALT                    ;SHOULD HAVE TRAPPED,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 762
1075 013036                          1$:
1076 013036  013737  177766  000502          MOV     CPUERR,RCPUER   ;SAVE CPU ERROR REGISTER
1077 013044  042737  177413  000502          BIC     #CERMSK,RCPUER  ;MASK OFF UNUSED CPU ERROR REG BITS
1078 013052  022737  000004  000502          CMP     #4,RCPUER       ;IS YELLOW ZONE BIT SET?
1079 013060  001405                          BEQ     3$
     013062                          2$:
     013062  012737  000236  000302          MOV     #236,$FATAL     ;MOVE TO MAILBOX #  *******  236  *******
     013070  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     013072  000000                          HALT                    ;INCORRECT CPU ERROR REGISTER CONTENTS, OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 743
1080 013074  005037  177766          3$:     CLR     CPUERR          ;CLEAR YELLOW ZONE BIT
```

```
1081                                    .SBTTL  TEST #60 - TEST FOR DEC OF R6 ON OVERFLOW TRAP
                                        ;*****************************************************************************
                                        ;TEST 60 - TEST FOR DEC OF R6 ON OVERFLOW TRAP
                                        ;*****************************************************************************
013100  032737  000001  177766  TST60:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013106  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013110  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013116  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013120  000000                          HALT                    ;CPU POWER BIT FOUND SET
013122  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013130  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013134  022737  000060  000304          CMP     #60,$TESTN      ;SEQUENCE ERROR?
013142  001011                          BNE     TST61-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1082 013144  012706  000150            MOV     #150,R6         ;R6 = 150
1083 013150  012737  013160  000004    MOV     #1$,4           ;TRAP POINTER
1084 013156  005746                    TST     -(R6)           ;WITH R6 = 150 SHOULD TRAP
1085 013160  020627  000142    1$:     CMP     R6,#142         ;DID R6 DECREMENT
1086 013164  001405                    BEQ     TST61
     013166  012737  000237  000302    MOV     #237,$FATAL     ;MOVE TO MAILBOX #  *******  237  *******
     013174  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     013176  000000                    HALT                    ;R6 NOT = 142,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

```
1087                                    .SBTTL  TEST #61 - TEST DIFFERENT TYPES OF OVERFLOW
                                        ;******************************************************************************
                                        ;TEST 61 - TEST DIFFERENT TYPES OF OVERFLOW
                                        ;******************************************************************************
013200  032737  000001  177766  TST61:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013206  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013210  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013216  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013220  000000                          HALT                    ;CPU POWER BIT FOUND SET
013222  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013230  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013234  022737  000061  000304          CMP     #61,$TESTN      ;SEQUENCE ERROR?
013242  001043                          BNE     TST62-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1088 013244  012706  000150            MOV     #150,R6
1089 013250  005037  000146            CLR     146             ;STATUS WORD OF LOC 10
1090 013254  012737  013264  000004    MOV     #1$,4           ;RETURN TO LOC 4
1091 013262  005246                    INC     -(6)
1092 013264  005737  000146    1$:     TST     146
1093 013270  001005                    BNE     2$
013272  012737  000240  000302          MOV     #240,$FATAL     ;MOVE TO MAILBOX #  *******  240  *******
013300  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013302  000000                          HALT                    ;INCREMENT OPERATION NOT INHIBITED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 757
1094 013304  012705  001000    2$:     MOV     #1000,R5
1095 013310  012706  000400            MOV     #400,R6
1096 013314  012737  013336  000004    MOV     #3$,4
1097 013322  124645                    CMPB    -(R6),-(R5)
1098 013324  012737  000241  000302    MOV     #241,$FATAL     ;MOVE TO MAILBOX #  *******  241  *******
013332  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013334  000000                          HALT                    ;STACK = 400 AND DECREMENTED, SHOULD TRAP
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 742
1099 013336  012706  000400    3$:     MOV     #400,R6
1100 013342  012737  013364  000004    MOV     #5$,4
1101 013350  134546                    BITB    -(R5),-(R6)
1102 013352                    4$:
013352  012737  000242  000302          MOV     #242,$FATAL     ;MOVE TO MAILBOX #  *******  242  *******
013360  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013362  000000                          HALT                    ;NO STACK OVERFLOW,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 727
1103 013364                    5$:
1113
```

```
1114                                    .SBTTL  TEST #62 - TEST THAT AN 10 CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************
                                        ;TEST 62 - TEST THAT AN 10 CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************
013364  032737  000001  177766  TST62:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013372  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013374  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013402  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013404  000000                          HALT                    ;CPU POWER BIT FOUND SET
013406  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013414  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013420  022737  000062  000304          CMP     #62,$TESTN      ;SEQUENCE ERROR?
013426  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
013430  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
013434  012737  013452  000010          MOV     #1$,10  ;SET UP 10 VECTOR
013442  012737  013464  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
013450  000010                          10                      ;THIS TRAP SHOULD CAUSE OVERFLOW
013452                          1$:
013452  012737  000243  000302          MOV     #243,$FATAL     ;MOVE TO MAILBOX #  *******  243  *******
013460  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013462  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761

013464  012737  000012  000010  2$:     MOV     #10+2,10
```

```
1115                                    .SBTTL  TEST #63 - TEST THAT AN IOT CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************
                                        ;TEST 63 - TEST THAT AN IOT CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************
013472  032737  000001  177766  TST63:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013500  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013502  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013510  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013512  000000                          HALT                    ;CPU POWER BIT FOUND SET
013514  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013522  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013526  022737  000063  000304          CMP     #63,$TESTN      ;SEQUENCE ERROR?
013534  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
013536  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
013542  012737  013560  000020          MOV     #1$,20  ;SET UP IOT VECTOR
013550  012737  013572  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
013556  000004                          IOT                     ;THIS TRAP SHOULD CAUSE OVERFLOW
013560                          1$:
013560  012737  000244  000302          MOV     #244,$FATAL     ;MOVE TO MAILBOX #  *******  244  *******
013566  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013570  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
013572  012737  000022  000020  2$:     MOV     #20+2,20
```

```
1116                                    .SBTTL  TEST #64 - TEST THAT AN EMT CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
                                        ;TEST 64 - TEST THAT AN EMT CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
013600  032737  000001  177766  TST64:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013606  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013610  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013616  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013620  000000                          HALT                    ;CPU POWER BIT FOUND SET
013622  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013630  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013634  022737  000064  000304          CMP     #64,$TESTN      ;SEQUENCE ERROR?
013642  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
013644  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
013650  012737  013666  000030          MOV     #1$,30  ;SET UP EMT VECTOR
013656  012737  013700  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
013664  104000                          EMT                     ;THIS TRAP SHOULD CAUSE OVERFLOW
013666                          1$:
013666  012737  000245  000302          MOV     #245,$FATAL     ;MOVE TO MAILBOX #  *******  245  *******
013674  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013676  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761

013700  012737  000032  000030  2$:     MOV     #30+2,30
```

TEST #65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP

```
1117                                    .SBTTL  TEST #65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
                                        ;TEST 65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
013706  032737  000001  177766  TST65: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013714  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013716  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013724  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013726  000000                        HALT                    ;CPU POWER BIT FOUND SET
013730  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013736  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013742  022737  000065  000304        CMP     #65,$TESTN      ;SEQUENCE ERROR?
013750  001011                        BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
013752  012706  000400                MOV     #400,R6         ;SET UP STACK TO OVERFLOW
013756  012737  013774  000034        MOV     #1$,34  ;SET UP TRAP VECTOR
013764  012737  014006  000004        MOV     #2$,4           ;SET UP OVERFLOW VECTOR
013772  104400                        TRAP                    ;THIS TRAP SHOULD CAUSE OVERFLOW
013774                          1$:
013774  012737  000246  000302        MOV     #246,$FATAL     ;MOVE TO MAILBOX #  *******  246  *******
014002  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014004  000000                        HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
014006  012737  000036  000034  2$:    MOV     #34+2,34
```

TEST #66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP

```
1118                                    .SBTTL  TEST #66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************************
                                        ;TEST 66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************************
014014  032737  000001  177766  TST66:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014022  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
014024  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
014032  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014034  000000                          HALT                    ;CPU POWER BIT FOUND SET
014036  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
014044  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
014050  022737  000066  000304          CMP     #66,$TESTN      ;SEQUENCE ERROR?
014056  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
014060  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
014064  012737  014102  000014          MOV     #1$,14  ;SET UP TRT VECTOR
014072  012737  014114  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
014100  000003                          TRT                     ;THIS TRAP SHOULD CAUSE OVERFLOW
014102                          1$:
014102  012737  000247  000302          MOV     #247,$FATAL     ;MOVE TO MAILBOX #  *******  247  *******
014110  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014112  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761

014114  012737  000016  000014  2$:     MOV     #14+2,14
```

```
1119                                    .SBTTL  TEST #67 - TEST THAT AN ILLA CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
                                        ;TEST 67 - TEST THAT AN ILLA CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
014122  032737  000001  177766  TST67:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014130  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
014132  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
014140  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014142  000000                          HALT                    ;CPU POWER BIT FOUND SET
014144  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
014152  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
014156  022737  000067  000304          CMP     #67,$TESTN      ;SEQUENCE ERROR?
014164  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
014166  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
014172  012737  014210  000010          MOV     #1$,10  ;SET UP ILLA VECTOR
014200  012737  014222  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
014206  004700                          ILLA                    ;THIS TRAP SHOULD CAUSE OVERFLOW
014210                          1$:
014210  012737  000250  000302          MOV     #250,$FATAL     ;MOVE TO MAILBOX #  *******  250  *******
014216  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014220  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761

014222  012737  000012  000010  2$:     MOV     #10+2,10
1120 014230  020627  000370             CMP     R6,#370         ;STACK PUSHED FOUR WORDS?
1121 014234  001405                     BEQ     TST70
014236  012737  000251  000302          MOV     #251,$FATAL     ;MOVE TO MAILBOX #  *******  251  *******
014244  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014246  000000                          HALT                    ;CORRECT # (4) OF WORDS WERE NOT PUSHED ONTO STACK
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 746
```

TEST #70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP

```
1122                                    .SBTTL  TEST #70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
                                        ;TEST 70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
014250  032737  000001  177766  TST70:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014256  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
014260  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
014266  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014270  000000                          HALT                    ;CPU POWER BIT FOUND SET
014272  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
014300  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
014304  022737  000070  000304          CMP     #70,$TESTN      ;SEQUENCE ERROR?
014312  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
014314  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
014320  012737  014336  000010          MOV     #1$,10  ;SET UP ILLB VECTOR
014326  012737  014350  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
014334  000100                          ILLB                    ;THIS TRAP SHOULD CAUSE OVERFLOW
014336                          1$:
014336  012737  000252  000302          MOV     #252,$FATAL     ;MOVE TO MAILBOX #  *******  252  *******
014344  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014346  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
014350  012737  000012  000010  2$:     MOV     #10+2,10
1123
```

```
1124                                    .SBTTL  TEST #71 - TEST FOR FALSE OVERFLOW TRAP
                                        ;*****************************************************************
                                        ;TEST 71 - TEST FOR FALSE OVERFLOW TRAP
                                        ;*****************************************************************
014356  032737  000001  177766  TST71:  BIT     #1,CPUER

                                        .REM %

                                                IDENTIFICATION

                                        PRODUCT CODE:           AC-F621D-MC
                                        PRODUCT NAME:           CKKABDO 11/44 TRAPS
                                        DATE CREATED:           OCTOBER, 1981
                                        MAINTAINER:             DIAGNOSTIC GROUP
                                        AUTHOR:                 DAN MILLEVILLE


                                        COPYRIGHT (C) 1979, 1981 DIGITAL EQUIPMENT CORP., MAYNARD, MASS.

                                        THIS SOFTWARE IS FURNISHED TO PURCHASER UNDER A LICENSE FOR USE
                                        ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION
                                        OF DEC'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT
                                        AS MAY OTHERWISE BE PROVIDED IN WRITING BY DEC.

                                        THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT
                                        NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
                                        EQUIPMENT CORPORATION.

                                        DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF
                                        ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.
```

```
                        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 , THE RESET AND WAIT INSTRUCTIONS.
```

```
                        2.      REQUIREMENTS

                        2.1     EQUIPMENT

                                11/44 STANDARD COMPUTER

                        2.2     STORAGE

                                2.2.1 PROGRAM STORAGE - THE ROUTINE USES MEMORY
                                      FROM 0000 TO 17600.
```

```
                        3.      LOADING PROCEDURE
                        3.1     METHOD
                                PROCEDURE FOR NORMAL ABSOLUTE TAPES SHOULD BE FOLLOWED.
```

```
                        4.      STARTING PROCEDURE


                                THE PROGRAM STARTS  AT 200.

                                IF IT IS DESIRED TO RESET THE PASS COUNT BACK TO
                                ZERO ; THEN START THIS PROGRAM AT LOCATION 210

                        4.1     PROGRAM AND/OR OPERATOR ACTION

                                LOAD PROGRAM INTO MEMORY. (BOTTOM 4K)
                                LOAD ADDRESS.
                                START.
                                THE PROGRAM WILL LOOP.
                                IT WILL PRINT ''CKKABD0 11/44 TRAPS'' AFTER THE FIRST ITERATION
                                AND THEN PRINTS IT EVERY 15 TIMES(APROXIMATELY EVERY 15 SECONDS)
```

```
                5.      OPERATION


                5.1     SUBROUTINE ABSTRACTS

                        5.1.1 BEGIN AT 200

                        5.1.2  SCOPE
                               -----
                               IF A SCOPE LOOP IS NEEDED INSERT A BRANCH AS THE
                               COMMENT TO THE HALT EXPLAINS.


                        5.1.3   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 PRINCIPLE OF THIS ROUTINE IS: THE VECTOR ENTRANCE
                                ADDRESS POINTS TO THE NEXT SEQUENTIAL WORD WHICH WILL CON-
                                TAIN 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
                                WHERE THE PROGRAM WAS. WHEN THE INTERRUPT OR
                                TRAP OCCURRED; MEMORY AS SPECIFIED BY R6 CONTAINS THE
                                PC OF THE INSTRUCTION FOLLOWING THE INSTRUCTION WHERE THE
                                TRAP OCCURRED.
                                THE CONTENTS OF LOCATION '$TESTN'(304) CONTAINS
                                THE TEST NUMBER THAT IT WAS DOING BEFORE IT
                                TRAPPED.

                5.2     PROGRAM AND/OR OPERATOR ACTION

                        5.2.1  LOADING AND STARTING AT 200 STARTS THE TEST. IF
                               AN ERROR IS DETECTED, THERE WILL BE A HALT.
                                NOTE:IF A SCOPE LOOP IS NEEDED
                               THE COMMENT SECTION OF THE HALT EXPLAINS
                                HOW TO UTILIZE THIS LOOP.
```

```
                        6.      ERRORS

                        6.1     ALL ERRORS WILL CAUSE A HALT.

                        6.1.1   THE PROGRAM CHECKS TO SEE THAT THE P.C. DOESN'T JUMP
                                WITHIN THE TESTS, BY A SEQUENCE COUNT CALLED '$TSTN'
                                THIS TEST IS A SEQUENCIAL INCREMENT AND COMPARE COUNT.

                                EXAMPLE

                        TSTA:   INC     $TSTNM          ;INCREMENT THE TEST NUMBER
                                CMP     #A,$TSTNM       ;COMPARE FOR THE RIGHT TEST
                                BNE     TSTA+1-12       ;IF NOT CORRECT BRANCH TO A HALT
                        **OR:
                                BEQ     1000$           ;BRANCH AROUND JUMP IF OK
                                JMP     TSTA+1-12       ;JUMP (USED FOR LONG TESTS THE BRANCH CAN'T REACH)
                        1000$:
                                ----
                                CODE



                                IMPORTANT
                        ******************************

                                IF AN ERROR IS DETECTED ;IT COULD BE BECAUSE OF THREE REASONS.
                                A)      WRONG TEST NUMBER
                                B)      ERROR IN THE PRESENT TEST.
                                C)      POWER MONITOR BIT PROBLEM.

                                ////////////////////////////////////////////////////////
                                THE TEST SEQUENCE LOCATION "TESTN" SHOULD BE CHECKED FIRST
                                TO SEE IF IT MATCHES THE PRESENT TEST.
                                IF IT DOESN'T MATCH ; THEN THE CONTENTS OF THIS LOCATION
                                TELL YOU WHICH TEST IT WAS DOING BEFORE IT HALTED.
                                ////////////////////////////////////////////////////////

                        6.2     ERROR RECOVERY

                                ON TRAP ERRORS - RESTART AT STARTING ADDRESS

                                ON POWER MONITOR ERRORS - FIND OUT WHAT POWER SUPPLY PROBLEM (OR CPU
                                ERROR REGISTER PROBLEM) THERE IS AND OBTAIN REPAIR BEFORE RERUNNING
                                THIS DIAGNOSTIC
```

```
159
160
161
162
163
164
165
166
167
```

7. RESTRICTIONS

7.1 STARTING RESTRICTION

NONE

7.2 OPERATIONAL RESTRICTION

NONE

```
168
169
170
171
172
```

8. MISCELLANEOUS

8.1 EXECUTION TIME

1ST PASS APPROX. 2 SEC., THEREAFTER EVERY 15 SEC

```
                        9.      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 RESERVED 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.
                                THE RTI AND RTT INSTRUCTIONS ARE CHECKED. STACK OVERFLOW IS
                                ALSO CHECKED FOR ALL THE TRAP INSTRUCTIONS.
                                SPECIAL CHECKS ARE MADE TO SEE IF BUS
                                ERROR TRAPS OCCUR ON NON-EXISTENT MEMORY.PIRQ TRAPS ARE CHECKED
                                AT ALL LEVELS.
```

```
                10.0    RUNNING UNDER APT

                        THE EXECUTION TIMES PROVIDED IN THE APT SCRIPT THAT FOLLOWS
                ARE FOR EXECUTION WITH A 11/44 PROCESSOR,  CACHE,
                16K CORE MEMORY, AND 300 BAUD.
                        THE FOLLOWING IS A PROGRAM LOAD FILE USED BY APT:
                1. E TABLE 'A' IS USED FOR APT DUMP MODE.

                2. E TABLE 'B' IS USED FOR APT QV AND RUN TIME MODES.
                   $ENVM=040 INDICATES THAT TYPEOUTS WILL BE SUPRESSED.


                        1ST PASS        LONGEST         ADDITIONAL
                        RUN TIME        TEST TIME       RUN TIME
                        5                   5               0


                        ......          E TABLES        .......

                                        A               B
                E-MODE/S-MODE           000/000         040/001
                ($ENVM/$ENV)

                SWITCH REGISTER 1       000000          000000
                ($SWREG)

                SWITCH REGISTER 2       000000          000000
                CPU TYPE/OPTIONS        00/0000         00/0000
```

```
                        11.0            REVISION HISTORY

                                REVISION        DATE            COMMENT

                                CKKABA          MARCH 1979      ORIGINAL RELEASE
                                CKKABB          NOVEMBER 1980   FIX DIAGNOSTIC HALT WITH CIS SWITCH IN
                                                                THE MAINTENANCE POSITION
                                CKKABC          APRIL 1981      INCLUSION OF CPU ERROR REGISTER BIT 0
                                                                CHECKING BEFORE EACH TEST.
                                CKKABD          OCTOBER 1981    CHECKING THAT WITH PIRQ AND UNIBUS
                                                                INTERRUPTS PENDING, UNIBUS SERVICE IS
                                                                1ST.

                        %
```

```
                        .TITLE  CKKABDO 11/44 TRAPS
                        ;ALL INSTRUCTIONS THAT ARE RESERVED
                        ;SHOULD TRAP TO LOCATION 10, AND THE
                        ;PC THAT POINTS TO THE TRAPPING INSTRUCTION
                        ;SHOULD BE PLACED ON THE STACK
                        ;LISTING
                                .LIST   ME
                                .NLIST  MC
                                .NLIST  MD
                                .NLIST  CND
000000                          .ENABLE ABS
                                .ENABLE AMA
000000                          R0      =%0
000001                          R1      =%1
000001                          LAST    =%1
000002                          R2      =%2
000003                          R3      =%3
000003                          TAB     =%3
000004                          R4      =%4
000005                          R5      =%5
000005                          FIRST   =%5
000006                          R6      =%6
000006                          SP      =%6
000007                          R7      =%7
000007                          PC      =%7
000000                          HLT=HALT
000003                          TRT=3
000004                          RTRAP5=4                ;ILLEGAL ADDRESSES
000014                          RTRAP4=14               ;FOR TRACE TRAP
000030                          RTRAP3=30               ;FOR EMULATOR TRAP
000020                          RTRAP2=20               ;FOR IOT TRAP
000034                          RTRAP1=34               ;FOR TRAP INST
177564                          TTCSR=177564
177560                          TRCSR=177560
177564                          TPS=177564
177566                          TPB=177566
000240                          BELL=240
000240                          NOP=240
177776                          STATUS=177776
000010                          TRAPA=10
000010                          RTRAP=10
004700                          ILLA=004700
000100                          ILLB=100
177776                          CC=177776
177766                          CPUERR=177766
177413                          CERMSK=177413
177776                          PSW=177776
                        .MCALL  .$APTHDR,.$APTBLS,.$ACT11
```

```
382 000000  000002                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000002  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000004  000006                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000006  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000010  000012                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000012  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000014  000016                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000016  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000020  000022                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000022  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000024  000026                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000026  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000030  000032                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000032  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PRFVIOUS ADDRESS
    000034  000036                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000036  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000040  000042                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000042  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000044  000046                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000046  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000050  000052                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000052  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000054  000056                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000056  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000060  000062                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000062  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000064  000066                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000066  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000070  000072                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000072  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000074  000076                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000076  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000100  000102                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000102  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000104  000106                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000106  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000110  000112                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000112  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000114  000116                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000116  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000120  000122                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000122  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000124  000126                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000126  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000130  000132                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000132  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000134  000136                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000136  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000140  000142                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000142  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000144  000146                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000146  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000150  000152                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000152  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000154  000156                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
    000156  000000                HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
    000160  000162                .WORD   .+2             ;ADDRESS OF NEXT LOCATION
```

```
000162  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000164  000166                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000166  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000170  000172                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000172  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000174  000176                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000176  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000200  000202                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000202  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000204  000206                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000206  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000210  000212                  .WORD    +2             ;ADDRESS OF NEXT LOCATION
000212  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000214  000216                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000216  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000220  000222                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000222  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000224  000226                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000226  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000230  000232                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000232  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000234  000236                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000236  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000240  000242                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000242  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000244  000246                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000246  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000250  000252                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000252  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000254  000256                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000256  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000260  000262                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000262  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000264  000266                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000266  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000270  000272                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000272  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000274  000276                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000276  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000300  000302                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000302  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000304  000306                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000306  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000310  000312                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000312  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000314  000316                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000316  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000320  000322                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000322  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000324  000326                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000326  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000330  000332                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000332  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000334  000336                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000336  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000340  000342                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000342  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
```

```
000344  000346                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000346  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000350  000352                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000352  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000354  000356                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000356  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000360  000362                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000362  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000364  000366                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000366  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000370  000372                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000372  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000374  000376                  .WORD   .+2             ;ADDRESS OF NEXT LOCATION
000376  000000                  HALT                    ;IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
```

```
                000200                          .=200
000200  000137  000510                          JMP     BEGIN
                000210                          .=210
000210  005037  000306                          CLR     $PASS
000214  000137  000510                          JMP     BEGIN
                000300                          .=300
                                        .SBTTL  ACT11 HOOKS
                                        ;*****************************************************************************
                                        ;HOOKS REQUIRED BY ACT11
                000300                          $SVPC=.                 ;SAVE PC
                000046                          .=46
        000046  023550                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
                000052                          .=52
        000052  000000                          .WORD   0               ;;2)SET LOC.52 TO ZERO
                000300                          .=$SVPC                 ;; RESTORE PC
```

```
393                                     .SBTTL  APT MAILBOX-ETABLE
                                        ;******************************************************************************
                                        .EVEN
000300                                  $MAIL:                          ;;APT MAILBOX
000300  000000                          $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
000302  000000                          $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
000304  000000                          $TESTN: .WORD   ATESTN          ;;TEST NUMBER
000306  000000                          $PASS:  .WORD   APASS           ;;PASS COUNT
000310  000000                          $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
000312  000000                          $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
000314  000000                          $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
000316  000000                          $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
000320                                  $ETABLE:                        ;;APT ENVIRONMENT TABLE
000320     000                          $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
000321     000                          $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
000322  000000                          $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
000324  000000                          $USWR:  .WORD   AUSWR           ;;USER SWITCHES
000326  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
```

APT PARAMETER BLOCK

```
    395                                 .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
                 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                 $HIBTS: .WORD   0        ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
          000332 000300                 $MBADR: .WORD   $MAIL    ;;ADDRESS OF APT MAILBOX (BITS 0-15)
          000334 000005                 $TSTM:  .WORD   5        ;;RUN TIM OF LONGEST    ST
          000336 000005                 $PASTM: .WORD   5        ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
          000340 000000                 $UNITM: .WORD   0        ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
          000342 000014                         .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```

APT PARAMETER BLOCK

```
    397           000304                  $TSTNM=$TESTN
    398           000302                  $ERROR=$FATAL
    399
    400           000500                          .=500
    401 000500    000000                  BUFF:   .WORD   0
    402 000502    000000                  RCPUER: .WORD   0
    403 000504    000250                  KTVEC:  .WORD   250
    404 000506    000252                  KTSTA:  .WORD   252
```

```
405 000510  012706  000500          BEGIN:  MOV     #500,SP         ;SET UP STACK POINTER
406 000514  012737  177777  023576          MOV     #-1,PASSPT      ;CLEAR THE ITERATION COUNTER
407 000522  023737  000042  023550          CMP     42,$ENDAD       ;SEE IF TITLE MESSAGE SHOULD BE PRINTED
408 000530  001403                          BEQ     RESTRT          ;BRANCH AROUND MESSAGE PRINTING IF SO
409 000532  004737  024216                  JSR     PC,PRTMSG       ;GO PRINT MESSAGE USING
410 000536  024006                          .WORD   TITLE           ;STARTING ADDRESS OF THIS MESSAGE
411 000540  005037  000300          RESTRT: CLR     $MSGTY
412 000544  012706  000500                  MOV     #500,R6
413 000550  012737  024132  000024          MOV     #PWRDWN,24      ;SET UP THE POWER DOWN VECTOR
414 000556  012737  000340  000026          MOV     #340,26         ;SET UP POWER DOWN PRIORITY
415 000564  012737  000006  000004          MOV     #6,4            ;SET UP TRAP VECTORS 4 & 6.
416 000572  005037  000006                  CLR     6
417 000576  012737  000012  000010          MOV     #12,10
418 000604  005037  000012                  CLR     12
419 000610  005037  000304                  CLR     $TSTNM
420 000614  005037  000302                  CLR     $ERROR
421 000620  012702  000300                  MOV     #$MSGTY,R2
422 000624  000412                          BR      TST1
```

```
                                        ;SPECIAL CASE OF ODD;.EVEN .BYTE AND REGISTER 6
000000                                  HERE=0

000626  000000                          K1:     .WORD   0
000630  000000                          K2:     .WORD   0
000632  000000                          K3:     .WORD   0
000634  000000                          K4:     .WORD   0
000636  000000                          K5:     .WORD   0
000640  000000                          K6:     .WORD   0
000642  052525                          K7:     .WORD   052525
000644  052400                          K10:    .WORD   052400
000646  000000                          K11:    .WORD   0
000650  000000                          K12:    .WORD   0
```

```
                                        .SBTTL  TEST #1 - TEST AUTO INC AND DEC OF R6 FOR WORD AND BYTES
                                        ;******************************************************************************
                                        ;TEST 1 - TEST AUTO INC AND DEC OF R6 FOR WORD AND BYTES
                                        ;******************************************************************************
000652  032737  000001  177766  TST1:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
000660  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
000662  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
000670  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
000672  000000                          HALT                    ;CPU POWER BIT FOUND SET
000674  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
000702  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
000706  022737  000001  000304          CMP     #1,$TESTN       ;SEQUENCE ERROR?
000714  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
000716  000137  001220                  JMP     TST2-12         ;JUMP TO ERROR HALT ON SEQ ERROR
000722                          1000$:
437 000722  005006                      CLR     R6
438 000724  112637  000000              MOVB    (R6)+,HERE      ;SIX SHOULD INCREMENT BY TWO
439 000730  020627  000002              CMP     R6,#2
440 000734  001405                      BEQ     1$
    000736  012737  000001  000302      MOV     #1,$FATAL       ;MOVE TO MAILBOX #  *******  1  *******
    000744  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    000746  000000                      HALT                    ;R6 DID NOT AUTO INCREMENT BY TWO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764
441
442 000750  012706  001000          1$:     MOV     #1000,R6
443 000754  114627  000000              MOVB    -(R6),#HERE     ;SHOULD DECREMENT BY TWO
444 000760  020627  000776              CMP     R6,#776
445 000764  001405                      BEQ     2$
    000766  012737  000002  000302      MOV     #2,$FATAL       ;MOVE TO MAILBOX #  *******  2  *******
    000774  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    000776  000000                      HALT                    ;R6 DID NOT AUTO DECREMENT BY 2
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 750
446
447 001000  005006                  2$:     CLR     R6
448 001002  112626                          MOVB    (R6)+,(R6)+     ;DOUBLES AUTO INCREMENT OF R6
449 001004  020627  000004              CMP     R6,#4
450 001010  001405                      BEQ     3$
    001012  012737  000003  000302      MOV     #3,$FATAL       ;MOVE TO MAILBOX #  *******  3  *******
    001020  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    001022  000000                      HALT                    ;WRONG AUTO INCREMENT OF R6
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 736
451
452 001024  005006                  3$:     CLR     R6
453 001026  005004                          CLR     R4
454 001030  122624                          CMPB    (R6)+,(R4)+     ;TEST INCREMENT OF R6
455 001032  020627  000002              CMP     R6,#2
456 001036  001405                      BEQ     4$
    001040  012737  000004  000302      MOV     #4,$FATAL       ;MOVE TO MAILBOX #  *******  4  *******
    001046  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    001050  000000                      HALT                    ;WRONG INCREMENT OF R6
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 723
457
458 001052  005006                  4$:     CLR     R6
```

```
    459 001054  005004                          CLR     R4
    460 001056  122426                          CMPB    (R4)+,(R6)+     ;TEST INCREMENT OF R6
    461 001060  020627  000002                  CMP     R6,#2
    462 001064  001405                          BEQ     5$
        001066  012737  000005  000302          MOV     #5,$FATAL       ;MOVE TO MAILBOX #  *******  5  *******
        001074  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        001076  000000                          HALT                    ;WRONG INCREMENT OF R6
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 710
    463
    464 001100  005006                  5$:     CLR     R6
    465 001102  005004                          CLR     R4
    466 001104  122624                          CMPB    (R6)+,(R4)+     ;TEST INCREMENT OF R4
    467 001106  020427  000001                  CMP     R4,#1
    468 001112  001405                          BEQ     6$
        001114  012737  000006  000302          MOV     #6,$FATAL       ;MOVE TO MAILBOX #  *******  6  *******
        001122  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        001124  000000                          HALT                    ;WRONG INCREMENT OF R4
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 675
    469
    470 001126  005006                  6$:     CLR     R6
    471 001130  005004                          CLR     R4
    472 001132  122426                          CMPB    (R4)+,(R6)+     ;TEST INCREMENT OF R6
    473 001134  020627  000002                  CMP     R6,#2
    474 001140  001405                          BEQ     7$
        001142  012737  000007  000302          MOV     #7,$FATAL       ;MOVE TO MAILBOX #  *******  7  *******
        001150  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        001152  000000                          HALT                    ;WRONG INCREMENT OF R6
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 662
    475
    476 001154  005006                  7$:     CLR     R6
    477 001156  005004                          CLR     R4
    478 001160  122426                          CMPB    (R4)+,(R6)+     ;TEST INCREMENT OF R4
    479 001162  020427  000001                  CMP     R4,#1
    480 001166  001405                          BEQ     8$
        001170  012737  000010  000302          MOV     #10,$FATAL      ;MOVE TO MAILBOX #  *******  10  *******
        001176  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        001200  000000                          HALT                    ;WRONG INCREMENT OF R4
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 647
    481
    482 001202  012706  001000          8$:     MOV     #1000,R6
    483 001206  124627  000000                  CMPB    -(R6),#HERE     ;TEST DECREMENT OF R6
    484 001212  022706  000776                  CMP     #776,R6
    485 001216  001405                          BEQ     TST2
        001220  012737  000011  000302          MOV     #11,$FATAL      ;MOVE TO MAILBOX #  *******  11  *******
        001226  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        001230  000000                          HALT                    ;WRONG DECREMENT OF R6,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 633
```

```
                                .SBTTL  TEST #2 - TEST TRANSFER OF .BYTE USING R6
                                ;*****************************************************************
                                ;TEST 2 - TEST TRANSFER OF .BYTE USING R6
                                ;*****************************************************************
001232  032737  000001  177766  TST2:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
001240  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
001242  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
001250  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001252  000000                          HALT                    ;CPU POWER BIT FOUND SET
001254  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
001262  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
001266  022737  000002  000304          CMP     #2,$TESTN       ;SEQUENCE ERROR?
001274  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
001276  000137  001600                  JMP     TST3-12         ;JUMP TO ERROR HALT ON SEQ ERROR
001302                          1000$:
487 001302  012737  123456  000636          MOV     #123456,K5
488 001310  012737  050505  000626          MOV     #050505,K1
489 001316  012705  000626                  MOV     #K1,R5          ;R5=(050505)K1
490 001322  012706  000636                  MOV     #K5,R6          ;R6=(123456)K5
491 001326  112625                          MOVB    (R6)+,(R5)+     ;LOW .BYTE OF R6 TO R5
492 001330  022737  050456  000626          CMP     #050456,K1
493 001336  001405                          BEQ     1$
001340  012737  000012  000302              MOV     #12,$FATAL      ;MOVE TO MAILBOX #  *******  12  *******
001346  005212                              INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001350  000000                              HALT                    ;FALSE TRANSFER OF .BYTE
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 753
494
495 001352  012737  123456  000636  1$:     MOV     #123456,K5
496 001360  012737  050505  000626          MOV     #050505,K1
497 001366  012705  000626                  MOV     #K1,R5          ;R5(050505)K1
498 001372  012706  000640                  MOV     #K6,R6          ;R6(123456)K5
499 001376  114625                          MOVB    -(R6),(R5)+     ;LOW .BYTE OF R6 TO R5 (DECREMENT)
500 001400  023727  000626  050456          CMP     K1,#050456
501 001406  001405                          BEQ     2$
001410  012737  000013  000302              MOV     #13,$FATAL      ;MOVE TO MAILBOX #  *******  13  *******
001416  005212                              INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001420  000000                              HALT                    ;FALSE R6 .BYTE TRANSFER
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 727
502
503 001422  012737  123456  000626  2$:     MOV     #123456,K1
504 001430  012737  050505  000636          MOV     #050505,K5
505 001436  012705  000626                  MOV     #K1,R5          ;(123456)
506 001442  012706  000636                  MOV     #K5,R6          ;(050505)
507 001446  112526                          MOVB    (R5)+,(R6)+     ;LOW OF R5 TO LOW OF R6
508 001450  022737  050456  000636          CMP     #050456,K5
509 001456  001405                          BEQ     3$
001460  012737  000014  000302              MOV     #14,$FATAL      ;MOVE TO MAILBOX #  *******  14  *******
001466  005212                              INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001470  000000                              HALT                    ;FALSE R6 .BYTE TRANSFER
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 703
510
511 001472  012737  123456  000626  3$:     MOV     #123456,K1
512 001500  012737  050505  000636          MOV     #050505,K5
513 001506  012705  000627                  MOV     #K1+1,R5        ;123456
```

TEST #2 - TEST TRANSFER OF .BYTE USING R6

```
    514 001512  012706  000636                  MOV     #K5,R6          ;050505
    515 001516  112526                          MOVB    (R5)+,(R6)+     ;HIGH OF R5 TO LOW OF R6
    516 001520  023727  000636  050647          CMP     K5,#050647
    517 001526  001405                          BEQ     4$
        001530  012737  000015  000302          MOV     #15,$FATAL      ;MOVE TO MAILBOX #  *******  15  *******
        001536  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        001540  000000                          HALT                    ;FALSE R6 .BYTE TRANSFER
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 657
    518
    519 001542  012737  123456  000626  4$:     MOV     #123456,K1
    520 001550  012737  050505  000636          MOV     #050505,K5
    521 001556  012705  000627                  MOV     #K1+1,R5        ;R5-123456-ODD ADDRESS
    522 001562  012706  000636                  MOV     #K5,R6          ;R6-050505--.EVEN ADDRESS
    523 001566  112625                          MOVB    (R6)+,(R5)+     ;LOW OF R6 TO HIGH OF R5
    524 001570  022737  042456  000626          CMP     #042456,K1
    525 001576  001405                          BEQ     TST3
        001600  012737  000016  000302          MOV     #16,$FATAL      ;MOVE TO MAILBOX #  *******  16  *******
        001606  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        001610  000000                          HALT                    ;FAILED LOW OF 6 TO HIGH OF 5,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 633
```

```
                                .SBTTL  TEST #3 - TEST BYTE OPERATION WITH SEQ ODD-EVEN ADDRESS
                                ;*****************************************************************
                                ;TEST 3 - TEST BYTE OPERATION WITH SEQ ODD-EVEN ADDRESS
                                ;*****************************************************************
001612  032737  000001  177766  TST3:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
001620  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
001622  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
001630  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001632  000000                          HALT                    ;CPU POWER BIT FOUND SET
001634  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
001642  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
001646  022737  000003  000304          CMP     #3,$TESTN       ;SEQUENCE ERROR?
001654  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
001656  000137  002070                  JMP     TST4-12         ;JUMP TO ERROR HALT ON SEQ ERROR
001662                          1000$:
527 001662  123737  000642  000643          CMPB    K7,K7+1         ;SAME .WORD LOW TC HIGH
528 001670  001405                          BEQ     5$
001672  012737  000017  000302          MOV     #17,$FATAL      ;MOVE TO MAILBOX #  *******  17  *******
001700  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001702  000000                          HALT                    ;SHOULD COMPARE LOW TO HIGH
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 766
529
530 001704  123737  000643  000642  5$:     CMPB    K7+1,K7         ;COMPARE ODD TO .EVEN SAME .WORD
531 001712  001405                          BEQ     6$
001714  012737  000020  000302          MOV     #20,$FATAL      ;MOVE TO MAILBOX #  *******  20  *******
001722  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001724  000000                          HALT                    ;ODD TO .EVEN .BYTE FAILURE
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
532
533 001726  123737  000645  000642  6$:     CMPB    K10+1,K7        ;SEQUENTIAL .BYTES
534 001734  001405                          BEQ     7$
001736  012737  000021  000302          MOV     #21,$FATAL      ;MOVE TO MAILBOX #  *******  21  *******
001744  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001746  000000                          HALT                    ;ODD TO .EVEN FAILED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 744
535
536 001750  123737  000644  000640  7$:     CMPB    K10,K6
537 001756  001405                          BEQ     8$
001760  012737  000022  000302          MOV     #22,$FATAL      ;MOVE TO MAILBOX #  *******  22  *******
001766  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
001770  000000                          HALT                    ;.EVEN TO EVEN FAILED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 733
538 001772  123737  000643  000645  8$:     CMPB    K7+1,K10+1
539 002000  001405                          BEQ     9$
002002  012737  000023  000302          MOV     #23,$FATAL      ;MOVE TO MAILBOX #  *******  23  *******
002010  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002012  000000                          HALT                    ;ODD TO ODD FAILED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 722
540
541 002014  123737  000644  000645  9$:     CMPB    K10,K10+1
542 002022  001005                          BNE     10$
002024  012737  000024  000302          MOV     #24,$FATAL      ;MOVE TO MAILBOX #  *******  24  *******
```

```
        002032  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        002034  000000                          HALT                    ;LOW TO HIGH IN SAME .WORD FAILED
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 711

    543
    544 002036  123737  000645  000645  10$:    CMPB    K10+1,K10+1
    545 002044  001405                          BEQ     11$
        002046  012737  000025  000302          MOV     #25,$FATAL      ;MOVE TO MAILBOX #  *******  25  *******
        002054  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        002056  000000                          HALT                    ;HIGH TO LOW IN SAME .WORD FAILED
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 700

    546
    547 002060  123737  000644  000643  11$:    CMPB    K10,K7+1
    548 002066  001005                          BNE     TST4
        002070  012737  000026  000302          MOV     #26,$FATAL      ;MOVE TO MAILBOX #  *******  26  *******
        002076  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        002100  000000                          HALT                    ;.EVEN TO ODD FAILED,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 667
```

TEST #4 - TEST THE CC BITS

```
549                                     .SBTTL  TEST #4 - TEST THE CC BITS
                                        ;******************************************************************************
                                        ;TEST 4 - TEST THE CC BITS
                                        ;******************************************************************************
002102  032737  000001  177766  TST4:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002110  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002112  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002120  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002122  000000                          HALT                    ;CPU POWER BIT FOUND SET
002124  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002132  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002136  022737  000004  000304          CMP     #4,$TESTN       ;SEQUENCE ERROR?
002144  001062                          BNE     TST5-12         ;BRANCH TO ERROR HALT ON SEQ ERROR
550 002146  000277                      SCC                     ;SET STATUS
551 002150  005037  177776              CLR     STATUS          ;CLEAR STATUS
552 002154  103005                      BCC     1$
002156  012737  000027  000302          MOV     #27,$FATAL      ;MOVE TO MAILBOX #  *******  27  *******
002164  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002166  000000                          HALT                    ;C NOT CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 766
553 002170                      1$:
002170  102005                          BVC     2$
002172  012737  000030  000302          MOV     #30,$FATAL      ;MOVE TO MAILBOX #  *******  30  *******
002200  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002202  000000                          HALT                    ;V NOT CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
554 002204                      2$:
002204  001005                          BNE     3$
002206  012737  000031  000302          MOV     #31,$FATAL      ;MOVE TO MAILBOX #  *******  31  *******
002214  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002216  000000                          HALT                    ;Z NOT CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 752
555 002220                      3$:
002220  100005                          BPL     4$
002222  012737  000032  000302          MOV     #32,$FATAL      ;MOVE TO MAILBOX #  *******  32  *******
002230  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002232  000000                          HALT                    ;N NOT CLEAR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 744
556 002234  000257              4$:     CCC                     ;CLEAR CONDITION CODES
557 002236  052737  000017  177776      BIS     #17,STATUS      ;SET STATUS TO ONES
558
559 002244  103405                      BCS     5$
002246  012737  000033  000302          MOV     #33,$FATAL      ;MOVE TO MAILBOX #  *******  33  *******
002254  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002256  000000                          HALT                    ;C NOT SET
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
560 002260                      5$:
002260  102405                          BVS     6$
002262  012737  000034  000302          MOV     #34,$FATAL      ;MOVE TO MAILBOX #  *******  34  *******
002270  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002272  000000                          HALT                    ;V NOT SET
                                                                ;TO SCOPE REPLACE HALT WITH 240
```

TEST #4 - TEST THE CC BITS

```
                                                                ;AND REPLACE NEXT INST WITH 724

561 002274                                      6$:
    002274  001405                                     BEQ     7$
    002276  012737  000035  000302                     MOV     #35,$FATAL      ;MOVE TO MAILBOX #  *******  35  *******
    002304  005212                                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002306  000000                                     HALT                    ;Z NOT SET
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 716

562 002310                                      7$:
    002310  100405                                     BMI     TST5
    002312  012737  000036  000302                     MOV     #36,$FATAL      ;MOVE TO MAILBOX #  *******  36  *******
    002320  005212                                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002322  000000                                     HALT                    ;N NOT SET,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 710
```

```
563                                     .SBTTL  TEST #5 - TEST THAT A TRAP OCCURS ON A RESERVED INS
                                        ;*****************************************************************
                                        ;TEST 5 - TEST THAT A TRAP OCCURS ON A RESERVED INS
                                        ;*****************************************************************
002324  032737  000001  177766  TST5:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002332  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002334  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002342  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002344  000000                          HALT                    ;CPU POWER BIT FOUND SET
002346  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002354  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002360  022737  000005  000304          CMP     #5,$TESTN       ;SEQUENCE ERROR?
002366  001006                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
564 002370  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
565 002374  012737  002416  000010      MOV     #2$,RTRAP       ;RETURN LOCATION
566 002402  000010                      TRAPA                   ;RESERVED INSTRUCTION, SHOULD TRAP
567 002404                      1$:
002404  012737  000037  000302          MOV     #37,$FATAL      ;MOVE TO MAILBOX #  *******  37  *******
002412  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002414  000000                          HALT                    ;RESERVE INSTRUCTION DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764
568 002416                      2$:
```

TEST #6 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

```
569                                     .SBTTL  TEST #6 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;**********************************************************************
                                        ;TEST 6 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;**********************************************************************
002416  032737  000001  177766  TST6:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002424  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002426  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002434  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002436  000000                          HALT                    ;CPU POWER BIT FOUND SET
002440  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002446  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002452  022737  000006  000304          CMP     #6,$TESTN       ;SEQUENCE ERROR?
002460  001011                          BNE     TST7-12         ;BRANCH TO ERROR HALT ON SEQ ERROR
570 002462  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
571 002466  012737  002476  000010      MOV     #1$,RTRAP       ;RETURN POINTER
572 002474  000010                      TRAPA                   ;RESERVED INSTRUCTION
573 002476  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
574 002502  001405                      BEQ     TST7
    002504  012737  000040  000302      MOV     #40,$FATAL      ;MOVE TO MAILBOX #  *******  40  *******
    002512  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002514  000000                      HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

```
                                .SBTTL  TEST #7 - TEST THAT PROPER P.C. IS SAVED
                                ;******************************************************************************
                                ;TEST 7 - TEST THAT PROPER P.C. IS SAVED
                                ;******************************************************************************
002516  032737  000001  177766  TST7:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002524  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002526  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002534  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002536  000000                          HALT                    ;CPU POWER BIT FOUND SET
002540  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002546  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002552  022737  000007  000304          CMP     #7,$TESTN       ;SEQUENCE ERROR?
002560  001012                          BNE     TST10-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
576 002562  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
577 002566  012737  002576  000010      MOV     #1$,RTRAP       ;RETURN FROM TRAP POINTER
578 002574  000010                      TRAPA                   ;TRAP ON THIS INSTRUCTION
579 002576  022737  002576  000474  1$:     CMP     #1$,BUFF-4      ;CHECK FOR INCREMENTED P.C.
580 002604  001405                      BEQ     TST10
002606  012737  000041  000302          MOV     #41,$FATAL      ;MOVE TO MAILBOX #  *******  41  *******
002614  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002616  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
```

```
                                        .SBTTL  TEST #10 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 10 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;
002620  032737  000001  177766  TST10:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002626  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
002630  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
002636  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
002640  000000                          HALT                    ;CPU POWER BIT FOUND SET
002642  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
002650  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
002654  022737  000010  000304          CMP     #10,$TESTN      ;SEQUENCE ERROR?
002662  001046                          BNE     TST11-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
582 002664  012706  000500              MOV     #BUFF,SP        ;SET UP
583 002670  012737  002706  000010      MOV     #1$,RTRAP       ;SET UP
584 002676  005037  177776              CLR     CC              ;CLEAR CC AND PRICRITY
585 002702  000257                      CCC
586 002704  000010                      TRAPA                   ;TRAP
587 002706  023727  000476  000000  1$:  CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
588 002714  001405                      BEQ     2$
    002716  012737  000042  000302      MOV     #42,$FATAL      ;MOVE TO MAILBOX #  *******  42  *******
    002724  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002726  000000                      HALT                    ;INCORRECT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
589 002730  012706  000500          2$:  MOV     #BUFF,SP        ;SET UP
590 002734  012737  002754  000010      MOV     #3$,RTRAP       ;SET UP
591 002742  012737  000357  177776      MOV     #357,CC         ;SET PRIORITY
592 002750  000277                      SCC                     ;SET CC
593 002752  000010                      TRAPA                   ;TRAP
594 002754  023727  000476  000357  3$:  CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
595 002762  000405                      BR      4$              ;BRANCH OVER ERROR #177
596 002764  012737  000043  000302      MOV     #43,$FATAL      ;MOVE TO MAILBOX #  *******  43  *******
    002772  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    002774  000000                      HALT                    ;THIS ERROR RESERVED FOR PWR MON BIT ERROR
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
597 002776                          4$:
    002776  001405                      BEQ     TST11
    003000  012737  000044  000302      MOV     #44,$FATAL      ;MOVE TO MAILBOX #  *******  44  *******
    003006  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003010  000000                      HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 724
```

```
                                .SBTTL  TEST #11 - TEST THAT 'NEW' STATUS IS CORRECT
                                ;*****************************************************************
                                ;TEST 11 - TEST THAT 'NEW' STATUS IS CORRECT
                                ;*****************************************************************
003012  032737  000001  177766  TST11:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET .
003020  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
003022  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
003030  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003032  000000                          HALT                    ;CPU POWER BIT FOUND SET
003034  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
003042  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
003046  022737  000011  000304          CMP     #11,$TESTN      ;SEQUENCE ERROR?
003054  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
003056  000137  003324                  JMP     12$             ;JUMP TO ERROR HALT ON SEQ ERROR
003062                          1000$:
599 003062  012706  000500              MOV     #BUFF,SP
600 003066  012737  003102  000010      MOV     #1$,RTRAP
601 003074  005037  000012              CLR     RTRAP+2         ;CLEAR FUTURE PRIORITY AND CC
602 003100  000010                      TPAPA
603 003102                      1$:                             ;TEST FOR "C" CLEARED
604 003102  100005                      BPL     2$
    003104  012737  000045  000302      MOV     #45,$FATAL      ;MOVE TO MAILBOX #  *******  45  *******
    003112  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003114  000000                      HALT                    ·N NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
605 003116                      2$:
    003116  001005                      BNE     3$
    003120  012737  000046  000302      MOV     #46,$FATAL      ;MOVE TO MAILBOX #  *******  46  *******
    003126  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003130  000000                      HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
606 003132                      3$:
    003132  102005                      BVC     4$
    003134  012737  000047  000302      MOV     #47,$FATAL      ;MOVE TO MAILBOX #  *******  47  *******
    003142  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003144  000000                      HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
607 003146                      4$:
    003146  103005                      BCC     5$
    003150  012737  000050  000302      MOV     #50,$FATAL      ;MOVE TO MAILBOX #  *******  50  *******
    003156  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003160  000000                      HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
608 003162  032737  000340  177776  5$:     BIT     #340,CC         ;TEST PRIORITY
609 003170  001405                      BEQ     6$
    003172  012737  000051  000302      MOV     #51,$FATAL      ;MOVE TO MAILBOX #  *******  51  *******
    003200  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003202  000000                      HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 726
610 003204  012706  000500          6$:     MOV     #BUFF,SP
611 003210  012737  003226  000010      MOV     #7$,RTRAP
612 003216  012737  000357  000012      MOV     #357,RTRAP+2    ;SET NEW "CC" AND PRIORITY
```

```
613 003224  000010                        TRAPA                   ;TRAP HERE
614 003226                          7$:
    003226  100405                        BMI     8$
    003230  012737  000052  000302        MOV     #52,$FATAL      ;MOVE TO MAILBOX #  *******  52  *******
    003236  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003240  000000                        HALT                    ;N NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 707

615 003242                          8$:
    003242  001405                        BEQ     9$
    003244  012737  000053  000302        MOV     #53,$FATAL      ;MOVE TO MAILBOX #  *******  53  *******
    003252  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003254  000000                        HALT                    ;Z NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 701

616 003256                          9$:
    003256  102405                        BVS     10$
    003260  012737  000054  000302        MOV     #54,$FATAL      ;MOVE TO MAILBOX #  *******  54  *******
    003266  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003270  000000                        HALT                    ;V NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 673

617 003272                          10$:
    003272  103405                        BCS     11$
    003274  012737  000055  000302        MOV     #55,$FATAL      ;MOVE TO MAILBOX #  *******  55  *******
    003302  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003304  000000                        HALT                    ;C NOT SET
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 665

618 003306  013706  177776          11$:  MOV     CC,SP
619 003312  042706  000017                BIC     #17,SP
620 003316  022706  000340                CMP     #340,SP
621 003322  001405                        BEQ     13$
    003324                          12$:
    003324  012737  000056  000302        MOV     #56,$FATAL      ;MOVE TO MAILBOX #  *******  56  *******
    003332  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003334  000000                        HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                  ;TO SCOPE REPLACE HALT WITH 240
                                                                  ;AND REPLACE NEXT INST WITH 651
622 003336  012737  000012  000010  13$:  MOV     #12,10
623 003344  005037  000012                CLR     12
```

```
                                .SBTTL  TEST #12 - TEST THAT A TRAP OCCURS FOR A "TRAP" INSTRUCTION
                                ;*************************************************************************
                                ;TEST 12 - TEST THAT A TRAP OCCURS FOR A "TRAP" INSTRUCTION
                                ;*************************************************************************
003350  032737  000001  177766  TST12:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003356  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
003360  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
003366  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003370  000000                         HALT                    ;CPU POWER BIT FOUND SET
003372  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
003400  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
003404  022737  000012  000304         CMP     #12,$TESTN      ;SEQUENCE ERROR?
003412  001013                         BNE     TST13-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
625 003414  012737  000012  000010     MOV     #12,10
626 003422  005037  000012             CLR     12
627 003426  012706  000500             MOV     #BUFF,SP        ;STACK POINTER SETUP
628 003432  012737  003454  000034     MOV     #1$,RTRAP1      ;RETURN LOCATION
629 003440  104400                     TRAP                    ;RESERVED INSTRUCTION, SHOULD TRAP
630 003442  012737  000057  000302     MOV     #57,$FATAL      ;MOVE TO MAILBOX #  *******  57  *******
003450  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003452  000000                         HALT                    ;TRAP DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 757

631 003454                      1$:
```

```
632                                     .SBTTL  TEST #13 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************
                                        ;TEST 13 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************
003454  032737  000001  177766  TST13:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003462  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
003464  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
003472  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003474  000000                          HALT                    ;CPU POWER BIT FOUND SET
003476  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
003504  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
003510  022737  000013  000304          CMP     #13,$TESTN      ;SEQUENCE ERROR?
003516  001011                          BNE     TST14-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
633 003520  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
634 003524  012737  003534  000034      MOV     #1$,RTRAP1      ;RETURN POINTER
635 003532  104400                      TRAP                    ;RESERVED INSTRUCTION
636 003534  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
637 003540  001405                      BEQ     TST14
003542  012737  000060  000302          MOV     #60,$FATAL      ;MOVE TO MAILBOX #  *******  60  *******
003550  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003552  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

```
638                                     .SBTTL  TEST #14 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
                                        ;TEST 14 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
    003554  032737  000001  177766  TST14:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
    003562  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
    003564  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
    003572  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003574  000000                          HALT                    ;CPU POWER BIT FOUND SET
    003576  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
    003604  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
    003610  022737  000014  000304          CMP     #14,$TESTN      ;SEQUENCE ERROR?
    003616  0C1012                          BNE     TST15-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
639 003620  012706  000500                  MOV     #BUFF,SP        ;STACK POINTER SETUP
640 003624  012737  003634  000034          MOV     #1$,RTRAP1      ;RETURN FROM TRAP POINTER
641 003632  104400                          TRAP                    ;TRAP ON THIS INSTRUCTION
642 003634  022737  003634  000474  1$:     CMP     #1$,BUFF-4      ;CHECK INCREMENTED P.C.
643 003642  001405                          BEQ     TST15
    003644  012737  000061  000302          MOV     #61,$FATAL      ;MOVE TO MAILBOX #  *******  61  *******
    003652  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    003654  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 760
```

```
644                                     .SBTTL  TEST #15 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 15 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
003656  032737  000001  177766  TST15:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003664  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
003666  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
003674  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003676  000000                          HALT                    ;CPU POWER BIT FOUND SET
003700  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
003706  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
003712  022737  000015  000304          CMP     #15,$TESTN      ;SEQUENCE ERROR?
003720  001037                          BNE     TST16-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
645 003722  012706  000500              MOV     #BUFF,SP        ;SET UP
646 003726  012737  003744  000034      MOV     #1$,RTRAP1      ;SET UP
647 003734  005037  177776              CLR     CC              ;CLEAR CC AND PRICRITY
648 003740  000257                      CCC
649 003742  104400                      TRAP                    ;TRAP
650 003744  023727  000476  000000  1$:  CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
651 003752  001405                      BEQ     2$
003754  012737  000062  000302          MOV     #62,$FATAL      ;MOVE TO MAILBOX #  *******  62  *******
003762  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
003764  000000                          HALT                    ;INCORRECT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
652 003766  012706  000500          2$:  MOV     #BUFF,SP        ;SET UP
653 003772  012737  004010  000034      MOV     #3$,RTRAP1      ;SET UP
654 004000  012737  000357  177776      MOV     #357,CC         ;SET PRIORITY
655 004006  104400                      TRAP                    ;SET CC
656 004010  023727  000476  000357  3$:  CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
657 004016  001405                      BEQ     TST16
00402C  012737  000063  000302          MOV     #63,$FATAL      ;MOVE TO MAILBOX #  *******  63  *******
004026  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004030  000000                          HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 733
```

TEST #16 - TEST THAT 'NEW' STATUS IS CORRECT

```
658                                     .SBTTL  TEST #16 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*****************************************************************
                                        ;TEST 16 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*****************************************************************
004032  032737  000001  177766  TST16:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004040  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
004042  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
004050  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004052  000000                          HALT                    ;CPU POWER BIT FOUND SET
004054  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
004062  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
004066  022737  000016  000304          CMP     #16,$TESTN      ;SEQUENCE ERROR?
004074  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
004076  000137  004344                  JMP     TST17-12                ;JUMP TO ERROR HALT ON SEQ ERROR
004102                          1000$:
659 004102  012706  000500              MOV     #BUFF,SP
660 004106  012737  004122  000034      MOV     #1$,RTRAP1
661 004114  005037  000036              CLR     RTRAP1+2        ;CLEAR FUTURE PRIORITY AND CC
662 004120  104400                      TRAP
663 004122                      1$:                             ;TEST FOR "C" CLEARED
664 004122  100005                      BPL     2$
004124  012737  000064  000302          MOV     #64,$FATAL      ;MOVE TO MAILBOX #  *******  64  *******
004132  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004134  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
665 004136                      2$:
004136  001005                          BNE     3$
004140  012737  000065  000302          MOV     #65,$FATAL      ;MOVE TO MAILBOX #  *******  65  *******
004146  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004150  000000                          HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
666 004152                      3$:
004152  102005                          BVC     4$
004154  012737  000066  000302          MOV     #66,$FATAL      ;MOVE TO MAILBOX #  *******  66  *******
004162  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004164  000000                          HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
667 004166                      4$:
004166  103005                          BCC     5$
004170  012737  000067  000302          MOV     #67,$FATAL      ;MOVE TO MAILBOX #  *******  67  *******
004176  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004200  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
668 004202  032737  000340  177776  5$:  BIT     #340,CC         ;TEST PRIORITY
669 004210  001405                      BEQ     6$
004212  012737  000070  000302          MOV     #70,$FATAL      ;MOVE TO MAILBOX #  *******  70  *******
004220  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004222  000000                          HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 726
670 004224  012706  000500          6$:  MOV     #BUFF,SP
671 004230  012737  004246  000034      MOV     #7$,RTRAP1
672 004236  012737  000357  000036      MOV     #357,RTRAP1+2   ;SET NEW "CC" AND PRIORITY
```

```
673 004244  104400                    TRAP                ;TRAP HERE
674 004246                        7$:
675 004246  100405                    BMI     8$
    004250  012737  000071  000302    MOV     #71,$FATAL  ;MOVE TO MAILBOX #  *******  71  * *****
    004256  005212                    INC     (R2)        ;SET MSGTYP TO FATAL ERROR
    004260  000000                    HALT                ;N NOT SET
                                                          ;TO SCOPE REPLACE HALT WITH 240
                                                          ;AND REPLACE NEXT INST WITH 707
676 004262                        8$:
    004262  001405                    BEQ     9$
    004264  012737  000072  000302    MOV     #72,$FATAL  ;MOVE TO MAILBOX #  *******  72  *******
    004272  005212                    INC     (R2)        ;SET MSGTYP TO FATAL ERROR
    004274  000000                    HALT                ;Z NOT SET
                                                          ;TO SCOPE REPLACE HALT WITH 240
                                                          ;AND REPLACE NEXT INST WITH 701
677 004276                        9$:
    004276  102405                    BVS     10$
    004300  012737  000073  000302    MOV     #73,$FATAL  ;MOVE TO MAILBOX #  *******  73  *******
    004306  005212                    INC     (R2)        ;SET MSGTYP TO FATAL ERROR
    004310  000000                    HALT                ;V NOT SET
                                                          ;TO SCOPE REPLACE HALT WITH 240
                                                          ;AND REPLACE NEXT INST WITH 673
678 004312                        10$:
    004312  103405                    BCS     11$
    004314  012737  000074  000302    MOV     #74,$FATAL  ;MOVE TO MAILBOX #  *******  74  *******
    004322  005212                    INC     (R2)        ;SET MSGTYP TO FATAL ERROR
    004324  000000                    HALT                ;C NOT SET
                                                          ;TO SCOPE REPLACE HALT WITH 240
                                                          ;AND REPLACE NEXT INST WITH 665
679 004326  013706  177776        11$:    MOV     CC,SP
680 004332  042706  000017            BIC     #17,SP
681 004336  022706  000340            CMP     #340,SP
682 004342  001405                    BEQ     TST17
    004344  012737  000075  000302    MOV     #75,$FATAL  ;MOVE TO MAILBOX #  *******  75  *******
    004352  005212                    INC     (R2)        ;SET MSGTYP TO FATAL ERROR
    004354  000000                    HALT                ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                          ;TO SCOPE REPLACE HALT WITH 240
                                                          ;AND REPLACE NEXT INST WITH 651
```

```
                                        .SBTTL  TEST #17 - TEST THAT ALL COMB  'TRAP' WILL CAUSE A TRAP
                                        ;**************************************************************
                                        ;TEST 17 - TEST THAT ALL COMB  'TRAP' WILL CAUSE A TRAP
                                        ;**************************************************************
 004356  032737  000001  177766  TST17:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 004364  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 004366  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 004374  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 004376  000000                          HALT                    ;CPU POWER BIT FOUND SET
 004400  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 004406  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 004412  022737  000017  000304          CMP     #17,$TESTN      ;SEQUENCE ERROR?
 004420  001011                          BNE     3$              ;BRANCH TO ERROR HALT ON SEQ ERROR
684 004422  012737  104400  004442          MOV     #TRAP,2$        ;INITIALIZE BASE TRAP INSTRUCTION
685 004430  012737  004456  000034          MOV     #4$,34          ;RETURN FROM TRAP TO RA1
686 004436  012706  000500          1$:     MOV     #BUFF,SP        ;SET UP STACK POINTER
687 004442  104400                  2$:     TRAP                    ;TRAP INST WILL BE MODIFIED TO TRAP+377
688 004444                          3$:
 004444  012737  000076  000302          MOV     #76,$FATAL      ;MOVE TO MAILBOX #  *******  76  *******
 004452  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 004454  000000                          HALT                    ;PREVIOUS INST FAILED TO TRAP,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 761
689 004456  005237  004442          4$:     INC     2$              ;INCREMENT TRAP INSTRUCTION
690 004462  022737  104777  004442          CMP     #104777,2$      ;TRAP+377 TO UPPER LIMIT
691 004470  103362                          BHIS    1$              ;HAVE WE TESTED ALL
692 004472  012737  000036  000034          MOV     #36,34
693 004500  005037  000036                  CLR     36
```

TEST #20 - TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION

```
694                                             .SBTTL  TEST #20 - TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION
                                                ;*****************************************************************
                                                ;TEST 20 - TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION
                                                ;*****************************************************************
004504  032737  000001  177766  TST20:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004512  001410                   BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
004514  012737  000177  000302           MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
004522  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004524  000000                   HALT                    ;CPU POWER BIT FOUND SET
004526  042737  000001  177766           BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
004534  005237  000304           100$:   INC     $TESTN          ;UPDATE TEST NUMBER
004540  022737  000020  000304           CMP     #20,$TESTN      ;SEQUENCE ERROR?
004546  001006                   BNE     TST21-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
695 004550  012706  000500           MOV     #BUFF,SP        ;STACK POINTER SETUP
696 004554  012737  004576  000020   MOV     #1$,RTRAP2      ;RETURN LOCATION
697 004562  000004                   IOT                     ;RESERVE INSTRUCTION, SHOULD TRAP
698 004564  012737  000077  000302   MOV     #77,$FATAL      ;MOVE TO MAILBOX #  *******  77  *******
004572  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004574  000000                   HALT                    ;IOT DIDN'T TRAP,OR WRONG $TSTNM
                                                ;TO SCOPE REPLACE HALT WITH 240
                                                ;AND REPLACE NEXT INST WITH 764

699 004576                       1$:
```

TEST #21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

```
700                                     .SBTTL  TEST #21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************
                                        ;TEST 21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************
004576  032737  000001  177766  TST21:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004604  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
004606  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
004614  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004616  000000                          HALT                    ;CPU POWER BIT FOUND SET
004620  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
004626  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
004632  022737  000021  000304          CMP     #21,$TESTN      ;SEQUENCE ERROR?
004640  001011                          BNE     TST22-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
701 004642  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
702 004646  012737  004656  000020      MOV     #1$,RTRAP2      ;RETURN POINTER
703 004654  000004                      IOT                     ;RESERVED INSTRUCTION
704 004656  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
705 004662  001405                      BEQ     TST22
004664  012737  000100  000302          MOV     #100,$FATAL     ;MOVE TO MAILBOX #  *******  100  *******
004672  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004674  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

TEST #22 - TEST THAT PROPER P.C. IS SAVED

```
706                                     .SBTTL  TEST #22 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************************
                                        ;TEST 22 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************************
004676  032737  000001  177766  TST22:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004704  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
004706  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
004714  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
004716  000000                          HALT                    ;CPU POWER BIT FOUND SET
004720  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
004726  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
004732  022737  000022  000304          CMP     #22,$TESTN      ;SEQUENCE ERROR?
004740  001012                          BNE     TST23-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
707 004742  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
708 004746  012737  004756  000020      MOV     #1$,RTRAP2      ;RETURN FROM TRAP POINTER
709 004754  000004                      IOT                     ;TRAP ON THIS INSTRUCTION
710 004756  022737  004756  000.74  1$:     CMP     #1$,BUFF-4      ;CHECK FOR INCREMENTED P.C.
711 004764  001405                          BEQ     TST23
    004766  012737  000101  000302          MOV     #101,$FATAL     ;MOVE TO MAILBOX #  *******  101  *******
    004774  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    004776  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 760
```

```
                                .SBTTL  TEST #23 - TEST THAT 'OLD" CC AND PRI ARE PLACED ON STACK
                                ;*****************************************************************
                                ;TEST 23 - TEST THAT "OLD" CC AND PRI ARE PLACED ON STACK
                                ;*****************************************************************
005000  032737  000001  177766  TST23:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005006  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
005010  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
005016  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005020  000000                         HALT                    ;CPU POWER BIT FOUND SET
005022  042737  000001  177766         BIC     #1,CPUFRR       ;CLEAR THE BIT FOUND SET
005030  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
005034  022737  000023  000304         CMP     #23,$TESTN      ;SEQUENCE ERROR?
005042  001040                         BNE     TST24-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
713 005044  012706  000500             MOV     #BUFF,SP        ;SET UP
714 005050  012737  005066  000020     MOV     #1$,RTRAP2      ;SET UP
715 005056  005037  177776             CLR     CC              ;CLEAR CC AND PRICRITY
716 005062  000257                     CCC
717 005064  000004                     IOT                     ;TRAP
718 005066  023727  000476  000000 1$:  CMP     BUFF 2,#0       ;TEST THAT OLD STATUS WENT TO STACK
719 005074  001405                     BEQ     2$
    005076  012737  0001C2  000302     MOV     #102,$FATAL     ;MOVE TO MAILBOX #  *******  102  *******
    005104  005212                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005106  000000                     HALT                    ;INCORRECT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
720 005110  012706  000500         2$:  MOV     #BUFF,SP        ;SET UP
721 005114  012737  005134  000020     MOV     #3$,RTRAP2      ;SET UP
722 005122  012737  000357  177776     MOV     #357,CC         ;SET PRIORITY
723 005130  000277                     SCC                     ;SET CC
724 005132  000004                     IOT                     ;TRAP
725 005134  023727  000476  000357 3$:  CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
726 005142  001405                     BEQ     TST24
    005144  012737  000103  000302     MOV     #103,$FATAL     ;MOVE TO MAILBOX #  *******  103  *******
    005152  005212                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005154  000000                     HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
```

```
                        .SBTTL  TEST #24 - TEST THAT "NEW" STATUS IS CORRECT
                        ;*****************************************************************
                        ;TEST 24 - TEST THAT "NEW" STATUS IS CORRECT
                        ;*****************************************************************
005156  032737  000001  177766  TST24:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005164  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
005166  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
005174  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005176  000000                          HALT                    ;CPU POWER BIT FOUND SET
005200  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
005206  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
005212  022737  000024  000304          CMP     #24,$TESTN      ;SEQUENCE ERROR?
005220  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
005222  000137  005470                  JMP     12$             ;JUMP TO ERROR HALT ON SEQ ERROR
005226                          1000$:
728 005226  012706  000500              MOV     #BUFF,SP
729 005232  012737  005246  000020      MOV     #1$,RTRAP2
730 005240  005037  000022              CLR     RTRAP2+2        ;CLEAR FUTURE PRIORITY AND CC
731 005244  000004                      IOT
732 005246                          1$:                         ;TEST FOR "C" CLEARED
733 005246  100005                      BPL     2$
    005250  012737  000104  000302      MOV     #104,$FATAL     ;MOVE TO MAILBOX #  *******  104  *******
    005256  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005260  000000                      HALT                    ;N NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
734 005262                          2$:
    005262  001005                      BNE     3$
    005264  012737  000105  000302      MOV     #105,$FATAL     ;MOVE TO MAILBOX #  *******  105  *******
    005272  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005274  000000                      HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
735 005276                          3$:
    005276  102005                      BVC     4$
    005300  012737  000106  000302      MOV     #106,$FATAL     ;MOVE TO MAILBOX #  *******  106  *******
    005306  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005310  000000                      HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
736 005312                          4$:
    005312  103005                      BCC     5$
    005314  012737  000107  000302      MOV     #107,$FATAL     ;MOVE TO MAILBOX #  *******  107  *******
    005322  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005324  000000                      HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
737 005326  032737  000340  177776  5$:  BIT     #340,CC         ;TEST PRIORITY
738 005334  001405                      BEQ     6$
    005336  012737  000110  000302      MOV     #110,$FATAL     ;MOVE TO MAILBOX #  *******  110  *******
    005344  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005346  000000                      HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 726
739 005350  012706  000500          6$:  MOV     #BUFF,SP
740 005354  012737  005372  000020      MOV     #7$,RTRAP2
741 005362  012737  000357  000022      MOV     #357,RTRAP2+2   ;SET NEW "CC" AND PRIORITY
```

TEST #24 - TEST THAT 'NEW' STATUS IS CORRECT

```
742 005370  000004                        IOT                     ;TRAP HERE
743 005372                          7$:
    005372  100405                        BMI     8$
    005374  012737  000111  000302        MOV     #111,$FATAL     ;MOVE TO MAILBOX #  *******  111  *******
    005402  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005404  000000                        HALT                    ;N NOT SET
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 707
744 005406                          8$:
    005406  001405                        BEQ     9$
    005410  012737  000112  000302        MOV     #112,$FATAL     ;MOVE TO MAILBOX #  *******  112  *******
    005416  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005420  000000                        HALT                    ;Z NOT SET
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 701
745 005422                          9$:
    005422  102405                        BVS     10$
    005424  012737  000113  000302        MOV     #113,$FATAL     ;MOVE TO MAILBOX #  *******  113  *******
    005432  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005434  000000                        HALT                    ;V NOT SET
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 673
746 005436                          10$:
    005436  103405                        BCS     11$
    005440  012737  000114  000302        MOV     #114,$FATAL     ;MOVE TO MAILBOX #  *******  114  *******
    005446  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005450  000000                        HALT                    ;C NOT SET
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 665
747 005452  013706  177776          11$:  MOV     CC,SP
748 005456  042706  000017                BIC     #17,SP
749 005462  022706  000340                CMP     #340,SP
750 005466  001405                        BEQ     13$
    005470                          12$:
    005470  012737  000115  000302        MOV     #115,$FATAL     ;MOVE TO MAILBOX #  *******  115  *******
    005476  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005500  000000                        HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                   ;TO SCOPE REPLACE HALT WITH 240
                                                                   ;AND REPLACE NEXT INST WITH 651
751 005502  012737  000022  000020  13$:  MOV     #22,20          ;.+2
752 005510  005037  000022                CLR     22              ;HALT
```

```
753                                     .SBTTL  TEST #25 - TEST THAT A TRAP OCCURS ON AN EMT  INS
                                        ;*************************************************************************
                                        ;TEST 25 - TEST THAT A TRAP OCCURS ON AN EMT  INS
                                        ;*************************************************************************
005514  032737  000001  177766  TST25:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005522  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
005524  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
005532  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005534  000000                          HALT                    ;CPU POWER BIT FOUND SET
005536  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
005544  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
005550  022737  000025  000304          CMP     #25,$TESTN      ;SEQUENCE ERROR?
005556  001006                          BNE     TST26-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
754 005560  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
755 005564  012737  005606  000030      MOV     #1$,RTRAP3      ;RETURN LOCATION
756 005572  104000                      EMT                     ;RESERVE INSTRUCTIOM, SHOULD TRAP
757 005574  012737  000116  000302      MOV     #116,$FATAL     ;MOVE TO MAILBOX #  *******  116  *******
005602  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005604  000000                          HALT                    ;EMT DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764
758 005606                          1$:
```

```
                                .SBTTL  TEST #26 - TEST DEC OF STACK POINTER ON A TRAP OPER
                                ;*************************************************************************
                                ;TEST 26 - TEST DEC OF STACK POINTER ON A TRAP OPER
                                ;*************************************************************************
005606  032737  000001  177766  TST26:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005614  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
005616  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
005624  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005626  000000                          HALT                    ;CPU POWER BIT FOUND SET
005630  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
005636  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
005642  022737  000026  000304          CMP     #26,$TESTN      ;SEQUENCE ERROR?
005650  001011                          BNE     TST27-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
760 005652  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
761 005656  012737  005666  000030      MOV     #1$,RTRAP3      ;RETURN POINTER
762 005664  104000                      EMT                     ;RESERVED INSTRUCTION
763 005666  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
764 005672  001405                      BEQ     TST27
005674  012737  000117  000302          MOV     #117,$FATAL     ;MOVE TO MAILBOX #  *******  117  *******
005702  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
005704  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

```
765                                     .SBTTL  TEST #27 - TEST THAT PROPER P.C. IS SAVED
                                        ;*************************************************************************
                                        ;TEST 27 - TEST THAT PROPER P.C. IS SAVED
                                        ;*************************************************************************
    005706  032737  000001  177766  TST27:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
    005714  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
    005716  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
    005724  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    005726  000000                          HALT                    ;CPU POWER BIT FOUND SET
    005730  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
    005736  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
    005742  022737  000027  000304          CMP     #27,$TESTN      ;SEQUENCE ERROR?
    005750  001012                          BNE     TST30-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
766 005752  012706  000500                  MOV     #BUFF,SP        ;STACK POINTER SETUP
767 005756  012737  005766  000030          MOV     #1$,RTRAP3      ;RTURN FROM TRAP POINTER
768 005764  104000                          EMT                     ;TRAP ON THIS INSTRICTION
769 005766  022737  005766  000474  1$:     CMP     #1$,BUFF-4      ;CHECK FOR INCREMENTED P.C.
770 005774  001405                          BEQ     TST30
    005776  012737  000120  000302          MOV     #120,$FATAL     ;MOVE TO MAILBOX #  *******  120  *******
    006004  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    006006  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 760
```

TEST #30 - TEST THAT 'OLD" CC AND PRI ARE PLACED ON STACK

```
                                .SBTTL  TEST #30 - TEST THAT 'OLD" CC AND PRI ARE PLACED ON STACK
                                ;*************************************************************************
                                ;TEST 30 - TEST THAT 'OLD" CC AND PRI ARE PLACED ON STACK
                                ;*************************************************************************
 006010  032737  000001  177766 TST30:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 006016  001410                 BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 006020  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 006026  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 006030  000000                 HALT                    ;CPU POWER BIT FOUND SET
 006032  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 006040  005237  000304        100$:    INC     $TESTN          ;UPDATE TEST NUMBER
 006044  022737  000030  000304         CMP     #30,$TESTN      ;SEQUENCE ERROR?
 006052  001040                 BNE     TST31-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 772 006054  012706  000500             MOV     #BUFF,SP        ;SET UP
 773 006060  012737  006076  000030     MOV     #1$,RTRAP3      ;SET UP
 774 006066  005037  177776             CLR     CC              ;CLEAR CC AND PRICRITY
 775 006072  000257                     CCC
 776 006074  104000                     EMT                     ;TRAP
 777 006076  023727  000476  000000 1$:  CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
 778 006104  001405                     BEQ     2$
     006106  012737  000121  000302     MOV     #121,$FATAL     ;MOVE TO MAILBOX #  *******  121  *******
     006114  005212                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     006116  000000                     HALT                    ;INCORRECT STATUS
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 755
 779 006120  012706  000500         2$:  MOV     #BUFF,SP        ;SET UP
 780 006124  012737  006144  000030     MOV     #3$,RTRAP3      ;SET UP
 781 006132  012737  000357  177776     MOV     #357,CC         ;SET PRIORITY
 782 006140  000277                     SCC                     ;SET CC
 783 006142  104000                     EMT                     ;TRAP
 784 006144  023727  000476  000357 3$:  CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
 785 006152  001405                     BEQ     TST31
     006154  012737  000122  000302     MOV     #122,$FATAL     ;MOVE TO MAILBOX #  *******  122  *******
     006162  005212                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     006164  000000                     HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 732
```

```
                                .SBTTL  TEST #31 - TEST THAT 'NEW" STATUS IS CORRECT
                                ;*************************************************************************
                                ;TEST 31 - TEST THAT 'NEW" STATUS IS CORRECT
                                ;*************************************************************************
006166  032737  000001  177766  TST31:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006174  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
006176  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
006204  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006206  000000                         HALT                    ;CPU POWER BIT FOUND SET
006210  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
006216  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
006222  022737  000031  000304         CMP     #31,$TESTN      ;SEQUENCE ERROR?
006230  001402                         BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
006232  000137  006474                 JMP     TST32-12                ;JUMP TO ERROR HALT ON SEQ ERROR
006236                          1000$:
787 006236  012706  000500             MOV     #BUFF,SP
788 006242  012737  006256  000030     MOV     #1$,RTRAP3
789 006250  005037  000032             CLR     RTRAP3+2        ;CLEAR FUTURE PRIORITY AND CC
790 006254  104000                     EMT
791 006256                      1$:                             ;TEST FOR "C" CLEARED
792 006256  100005                     BPL     2$
006260  012737  000123  000302         MOV     #123,$FATAL     ;MOVE TO MAILBOX #  *******  123  *******
006266  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006270  000000                         HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
793 006272                      2$:
006272  001005                         BNE     3$
006274  012737  000124  000302         MOV     #124,$FATAL     ;MOVE TO MAILBOX #  *******  124  *******
006302  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006304  000000                         HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
794 006306                      3$:
006306  102005                         BVC     4$
006310  012737  000125  000302         MOV     #125,$FATAL     ;MOVE TO MAILBOX #  *******  125  *******
006316  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006320  000000                         HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
795 006322                      4$:
006322  103005                         BCC     5$
006324  012737  000126  000302         MOV     #126,$FATAL     ;MOVE TO MAILBOX #  *******  126  *******
006332  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006334  000000                         HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
796 006336  032737  000340  177776  5$:     BIT     #340,CC         ;TEST PRIORITY
797 006344  001405                         BEQ     6$
006346  012737  000127  000302         MOV     #127,$FATAL     ;MOVE TO MAILBOX #  *******  127  *******
006354  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006356  000000                         HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 726
798 006360  012706  000500             MOV     #BUFF,SP
799 006364  012737  006402  000030     MOV     #7$,RTRAP3
800 006372  012737  000357  000032     MOV     #357,RTRAP3+2   ;SET NEW "CC" AND PRIORITY
```

TEST #31 - TEST THAT 'NEW' STATUS IS CORRECT

```
801 006400  104000                          EMT                     ;TRAP HERE
802 006402                          7$:
    006402  100405                          BMI     8$
    006404  012737  000130  000302          MOV     #130,$FATAL     ;MOVE TO MAILBOX #  *******  130  *******
    006412  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    006414  000000                          HALT                    ;N NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 707

803 006416                          8$:
    006416  001405                          BEQ     9$
    006420  012737  000131  000302          MOV     #131,$FATAL     ;MOVE TO MAILBOX #  *******  131  *******
    006426  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    006430  000000                          HALT                    ;Z NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 701

804 006432                          9$:
    006432  102405                          BVS     10$
    006434  012737  000132  000302          MOV     #132,$FATAL     ;MOVE TO MAILBOX #  *******  132  *******
    006442  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    006444  000000                          HALT                    ;V NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 673

805 006446                          10$:
    006446  103405                          BCS     11$
    006450  012737  000133  000302          MOV     #133,$FATAL     ;MOVE TO MAILBOX #  *******  133  *******
    006456  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    006460  000000                          HALT                    ;C NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 665

806 006462  000257                  11$:    CCC
807 006464  022737  000340  177776          CMP     #340,CC
808 006472  001405                          BEQ     TST32
    006474  012737  000134  000302          MOV     #134,$FATAL     ;MOVE TO MAILBOX #  *******  134  *******
    006502  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    006504  000000                          HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 653
```

```
809                                     .SBTTL  TEST #32 - TEST THAT ALL COMB  EMT WILL CAUSE A TRAP
                                        ;*************************************************************************
                                        ;TEST 32 - TEST THAT ALL COMB  EMT WILL CAUSE A TRAP
                                        ;*************************************************************************
     006506  032737  000001  177766  TST32:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     006514  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     006516  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     006524  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     006526  000000                          HALT                    ;CPU POWER BIT FOUND SET
     006530  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     006536  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     006542  022737  000032  000304          CMP     #32,$TESTN      ;SEQUENCE ERROR?
     006550  001011                          BNE     3$              ;BRANCH TO ERROR HALT ON SEQ ERROR
810  006552  012737  104000  006572          MOV     #EMT,2$         ;INITIALIZE BASE EMT INSTRUCTION
811  006560  012737  006606  000030          MOV     #4$,30          ;RETURN FROM TRAP TO 3$
812  006566  012706  000500          1$:     MOV     #BUFF,SP        ;SET UP STACK POINTER
813  006572  104000                  2$:     EMT                     ;TRAP INST. WILL BE MODIFIED TO EMT+377
814  006574                          3$:
     006574  012737  000135  000302          MOV     #135,$FATAL     ;MOVE TO MAILBOX #  *******  135  *******
     006602  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     006604  000000                          HALT                    ;PREVIOUS INST FAILED TO TRAP,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 761
815  006606  005237  006572          4$:     INC     2$              ;INCREMENT TRAP INSTRUCTION
816  006612  022737  104377  006572          CMP     #EMT+377,2$     ;EMT+377 TO EMT?
817  006620  103362                          BHIS    1$              ;HAVE WE TESTED ALL
818                                                                  ;YES
819  006622  012737  000032  000030          MOV     #32,30          ;/.+
820  006630  005037  000032                  CLR     32              ;HALT
```

TEST #33 - TEST THAT A TRAP OCCURES ON AN ''TRACE-TRT'' INS

```
821                                     .SBTTL  TEST #33 - TEST THAT A TRAP OCCURES ON AN ''TRACE-TRT'' INS
                                        ;*****************************************************************
                                        ;TEST 33 - TEST THAT A TRAP OCCURS ON AN ''TRACE-TRT'' INS
                                        ;*****************************************************************
006634  032737  000001  177766  TST33:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006642  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
006644  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
006652  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006654  000000                          HALT                    ;CPU POWER BIT FOUND SET
006656  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
006664  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
006670  022737  000033  000304          CMP     #33,$TESTN      ;SEQUENCE ERROR?
006676  001006                          BNE     TST34-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
822 006700  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
823 006704  012737  006726  000014      MOV     #1$,RTRAP4      ;RETURN LOCATION
824 006712  000003                      TRT                     ;RESERVED INSTRUCTION, SHOULD TRAP
825 006714  012737  000136  000302      MOV     #136,$FATAL     ;MOVE TO MAILBOX #  *******  136  *******
006722  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006724  000000                          HALT                    ;TRT DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764
826 006726                      1$:
```

```
827                                     .SBTTL  TEST #34 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************
                                        ;TEST 34 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*****************************************************************
006726  032737  000001  177766  TST34:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006734  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
006736  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
006744  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
006746  000000                          HALT                    ;CPU POWER BIT FOUND SET
006750  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
006756  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
006762  022737  000034  000304          CMP     #34,$TESTN      ;SEQUENCE ERROR?
006770  001011                          BNE     TST35-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
828 006772  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
829 006776  012737  007006  000014      MOV     #1$,RTRAP4      ;RETURN POINTER
830 007004  000003                      TRT                     ;RESERVED INSTRUCTION
831 007006  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
832 007012  001405                          BEQ     TST35
    007014  012737  000137  000302          MOV     #137,$FATAL     ;MOVE TO MAILBOX #  *******  137  *******
    007022  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007024  000000                          HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 761
```

TEST #35 - TEST THAT PROPER P.C. IS SAVED

```
                                        .SBTTL  TEST #35 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
                                        ;TEST 35 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
007026  032737  000001  177766  TST35:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007034  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
007036  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
007044  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007046  000000                          HALT                    ;CPU POWER BIT FOUND SET
007050  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
007056  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
007062  022737  000035  000304          CMP     #35,$TESTN      ;SEQUENCE ERROR?
007070  001012                          BNE     TST36-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
834 007072  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
835 007076  012737  007106  000014      MOV     #1$,RTRAP4      ;RETURN FROM TRAP POINTER
836 007104  000003                      TRT                     ;TRAP ON THIS INSTRUCTION
837 007106  022737  007106  000474  1$:  CMP     #.,BUFF-4       ;CHECK FOR INCREMENTED P.C.
838 007114  001405                      BEQ     TST36
    007116  012737  000140  000302      MOV     #140,$FATAL     ;MOVE TO MAILBOX #  *******  140  *******
    007124  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007126  000000                      HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ,AND REPLACE NEXT INST WITH 760
```

TEST #36 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK

```
839                                     .SBTTL  TEST #36 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 36 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
 007130  032737  000001  177766  TST36: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 007136  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 007140  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 007146  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 007150  000000                        HALT                    ;CPU POWER BIT FOUND SET
 007152  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 007160  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 007164  022737  000036  000304        CMP     #36,$TESTN      ;SEQUENCE ERROR?
 007172  001040                        BNE     TST37-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
840 007174  012706  000500            MOV     #BUFF,SP        ;SET UP
841 007200  012737  007216  000014    MOV     #1$,RTRAP4      ;SET UP
842 007206  005037  177776            CLR     CC              ;CLEAR CC AND PRICRITY
843 007212  000257                    CCC
844 007214  000003                    TRT                     ;TRAP
845 007216  023727  000476  000000  1$:   CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
846                                                             ;TEST FOR ALL ZEROS
847 007224  001405                    BEQ     2$
 007226  012737  000141  000302        MOV     #141,$FATAL     ;MOVE TO MAILBOX #  *******  141  *******
 007234  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 007236  000000                        HALT                    ;INCORRECT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
848 007240  012706  000500        2$:     MOV     #BUFF,SP        ;SET UP
849 007244  012737  007264  000014        MOV     #3$,RTRAP4      ;SET UP
850 007252  012737  000357  177776        MOV     #357,CC         ;SET PRIORITY
851 007260  000277                        SCC                     ;SET-SET CC
852 007262  000003                        TRT                     ;TRAP
853 007264  023727  000476  000357  3$:   CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
854 007272  001405                        BEQ     TST37
 007274  012737  000142  000302        MOV     #142,$FATAL     ;MOVE TO MAILBOX #  *******  142  *******
 007302  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 007304  000000                        HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
```

```
855                                             .SBTTL  TEST #37 - TEST THAT 'NEW' STATUS IS CORRECT
                                                ;*************************************************************
                                                ;TEST 37 - TEST THAT 'NEW' STATUS IS CORRECT
                                                ;*************************************************************
007306  032737  000001  177766  TST37:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007314  001410                   BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
007316  012737  000177  000302           MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
007324  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007326  000000                   HALT                    ;CPU POWER BIT FOUND SET
007330  042737  000001  177766           BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
007336  005237  000304   100$:   INC     $TESTN          ;UPDATE TEST NUMBER
007342  022737  000037  000304           CMP     #37,$TESTN      ;SEQUENCE ERROR?
007350  001402                   BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
007352  000137  007620                   JMP     15$             ;JUMP TO ERROR HALT ON SEQ ERROR
007356                           1000$:
856 007356  012706  000500           MOV     #BUFF,SP
857 007362  012737  007376  000014   MOV     #4$,RTRAP4
858 007370  005037  000016           CLR     RTRAP4+2        ;CLEAR FUTURE PRIORITY AND CC
859 007374  000003                   TRT
860 007376                       4$:                         ;TEST FOR "C" CLEARED
861 007376  100005                   BPL     5$
007400  012737  000143  000302       MOV     #143,$FATAL     ;MOVE TO MAILBOX #  *******  143  *******
007406  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007410  000000                   HALT                    ;C NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 761
862 007412                       5$:
007412  001005                   BNE     6$
007414  012737  000144  000302       MOV     #144,$FATAL     ;MOVE TO MAILBOX #  *******  144  *******
007422  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007424  000000                   HALT                    ;Z NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 753
863 007426                       6$:
007426  102005                   BVC     7$
007430  012737  000145  000302       MOV     #145,$FATAL     ;MOVE TO MAILBOX #  *******  145  *******
007436  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007440  000000                   HALT                    ;V NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 745
864 007442                       7$:
007442  103005                   BCC     8$
007444  012737  000146  000302       MOV     #146,$FATAL     ;MOVE TO MAILBOX #  *******  146  *******
007452  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007454  000000                   HALT                    ;C NOT CLEARED
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 737
865 007456  032737  000340  177776   8$:     BIT     #340,CC         ;TEST PRIORITY
866 007464  001405                   BEQ     9$
007466  012737  000147  000302       MOV     #147,$FATAL     ;MOVE TO MAILBOX #  *******  147  *******
007474  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007476  000000                   HALT                    ;PRIORITY NOT ZERO
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 726
867 007500  012706  000500           9$:     MOV     #BUFF,SP
868 007504  012737  007522  000014   MOV     #10$,RTRAP4
869 007512  012737  000357  000016   MOV     #357,RTRAP4+2   ;SET NEW "CC" AND PRIORITY
```

TEST #37 - TEST THAT 'NEW' STATUS IS CORRECT

```
870 007520  000003                    TRT                     ;TRAP HERE
871 007522                        10$:
    007522  100405                    BMI     11$
    007524  012737  000150  000302    MOV     #150,$FATAL     ;MOVE TO MAILBOX #  *******  150  *******
    007532  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007534  000000                    HALT                    ;N NOT SET
                                                               ;TO SCOPE REPLACE HALT WITH 240
                                                               ;AND REPLACE NEXT INST WITH 707
872 007536                        11$:
    007536  001405                    BEQ     12$
    007540  012737  000151  000302    MOV     #151,$FATAL     ;MOVE TO MAILBOX #  *******  151  *******
    007546  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007550  000000                    HALT                    ;Z NOT SET
                                                               ;TO SCOPE REPLACE HALT WITH 240
                                                               ;AND REPLACE NEXT INST WITH 701
873 007552                        12$:
    007552  102405                    BVS     13$
    007554  012737  000152  000302    MOV     #152,$FATAL     ;MOVE TO MAILBOX #  *******  152  *******
    007562  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007564  000000                    HALT                    ;V NOT SET
                                                               ;TO SCOPE REPLACE HALT WITH 240
                                                               ;AND REPLACE NEXT INST WITH 673
874 007566                        13$:
    007566  103405                    BCS     14$
    007570  012737  000153  000302    MOV     #153,$FATAL     ;MOVE TO MAILBOX #  *******  153  *******
    007576  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007600  000000                    HALT                    ;C NOT SET
                                                               ;TO SCOPE REPLACE HALT WITH 240
                                                               ;AND REPLACE NEXT INST WITH 665
875 007602  013706  177776        14$:    MOV     CC,SP
876 007606  042706  000017            BIC     #17,SP
877 007612  022706  000340            CMP     #340,SP
878 007616  001405                    BEQ     16$
    007620                        15$:
    007620  012737  000154  000302    MOV     #154,$FATAL     ;MOVE TO MAILBOX #  *******  154  *******
    007626  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    007630  000000                    HALT                    ;PRIORITY WAS C ANGED,OR WRONG $TSTNM
                                                               ;TO SCOPE REPLACE HALT WITH 240
                                                               ;AND REPLACE NEXT INST WITH 651
879 007632  012737  000016  000014  16$:    MOV     #16,14
880 007640  005037  000016            CLR     16
881
882                                ;PDP-11 ILLEGAL AND ADDRESS INSTRUCTION TEST
883                                ;ALL INSTRUCTIONS THAT ARE ILLEGAL
884                                ;SHOULD TRAP TO LOCATION 10, AND THE
885                                ;PC THAT POINTS TO THE TRAPPING INSTRUCTION
886                                ;SHOULD BE PLACED ON THE STACK
887
```

```
                                .SBTTL  TEST #40 - TEST THAT A TRAP OCCURS ON AN ILLEGAL INS
                                ;*************************************************************
                                ;TEST 40 - TEST THAT A TRAP OCCURS ON AN ILLEGAL INS
                                ;*************************************************************
 007644  032737  000001  177766 TST40:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 007652  001410                 BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 007654  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 007662  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 007664  000000                 HALT                    ;CPU POWER BIT FOUND SET
 007666  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 007674  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 007700  022737  000040  000304         CMP     #40,$TESTN      ;SEQUENCE ERROR?
 007706  001006                 BNE     TST41-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 007710  012706  000500         MOV     #BUFF,SP        ;STACK POINTER SETUP
 007714  012737  007736  000010         MOV     #1$,RTRAP       ;RETURN LOCATION
 007722  000100                 JMP     R0              ;ILLEGAL INSTRUCTION, SHOULD TRAP
 007724  012737  000155  000302         MOV     #155,$FATAL     ;MOVE TO MAILBOX #  *******  155  *******
 007732  005212                 INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 007734  000000                 HALT                    ;ILLEGAL INSTRUCTION DIDN'T TRAP,OR WRONG $TSTNM
                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                        ;AND REPLACE NEXT INST WITH 764
 007736                         1$:
```

```
894                                     .SBTTL  TEST #41 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*************************************************************************
                                        ;TEST 41 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*************************************************************************
007736  032737  000001  177766  TST41:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007744  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
007746  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
007754  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
007756  000000                          HALT                    ;CPU POWER BIT FOUND SET
007760  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
007766  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
007772  022737  000041  000304          CMP     #41,$TESTN      ;SEQUENCE ERROR?
010000  001011                          BNE     TST42-12                ;BRANCH TC ERROR HALT ON SEQ ERROR
895 010002  012706  000500              MOV     #BUFF,SP        ;STACK POINTER SETUP
896 010006  012737  010016  000010      MOV     #1$,RTRAP       ;RETURN POINTER
897 010014  000100                      JMP     R0              ;RESERVED INSTRUCTION
898 010016  020627  000474          1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
899 010022  001405                      BEQ     TST42
    010024  012737  000156  000302      MOV     #156,$FATAL     ;MOVE TO MAILBOX #  *******  156  *******
    010032  005212                      INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010034  000000                      HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

```
900                                     .SBTTL  TEST #42 - TEST THAT PROPER P.C. IS SAVED
                                        ;**************************************************************************
                                        ;TEST 42 - TEST THAT PROPER P.C. IS SAVED
                                        ;**************************************************************************
 010036  032737  000001  177766  TST42: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 010044  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 010046  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 010054  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010056  000000                        HALT                    ;CPU POWER BIT FOUND SET
 010060  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 010066  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 010072  022737  000042  000304        CMP     #42,$TESTN      ;SEQUENCE ERROR?
 010100  001012                        BNE     TST43-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
901 010102  012706  000500            MOV     #BUFF,SP        ;STACK POINTER SETUP
902 010106  012737  010116  000010    MOV     #1$,RTRAP       ;RETURN FROM TRAP POINTER
903 010114  000100                    JMP     R0              ;TRAP ON THIS INSTRUCTION
904 010116  022737  010116  000474  1$:   CMP     #.,BUFF-4       ;CHECK FOR INCREMENTED P.C.
905 010124  001405                    BEQ     TST43
 010126  012737  000157  000302        MOV     #157,$FATAL     ;MOVE TO MAILBOX #  *******  157  *******
 010134  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010136  000000                        HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
```

```
                                .SBTTL  TEST #43 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                ;*****************************************************************
                                ;TEST 43 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                ;*****************************************************************
 010140  032737  000001  177766 TST43:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 010146  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 010150  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 010156  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010160  000000                        HALT                    ;CPU POWER BIT FOUND SET
 010162  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 010170  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 010174  022737  000043  000304        CMP     #43,$TESTN      ;SEQUENCE ERROR?
 010202  001040                        BNE     TST44-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 010204  012706  000500                MOV     #BUFF,SP        ;SET UP
 010210  012737  010226  000010        MOV     #1$,RTRAP       ;SET UP
 010216  005037  177776                CLR     CC              ;CLEAR CC AND PRICRITY
 010222  000257                        CCC
 010224  000100                        JMP     R0              ;TRAP
 010226  023727  000476  000000 1$:     CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
 010234  001405                        BEQ     2$
 010236  012737  000160  000302        MOV     #160,$FATAL     ;MOVE TO MAILBOX #  *******  160  *******
 010244  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010246  000000                        HALT                    ;INCORRECT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
 010250  012706  000500        2$:     MOV     #BUFF,SP        ;SET UP
 010254  012737  010274  000010        MOV     #3$,RTRAP       ;SET UP
 010262  012737  000357  177776        MOV     #357,CC         ;SET PRIORITY
 010270  000277                        SCC                     ;SET CC
 010272  000100                        JMP     R0              ;TRAP
 010274  023727  000476  000357 3$:     CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
 010302  001405                        BEQ     TST44
 010304  012737  000161  000302        MOV     #161,$FATAL     ;MOVE TO MAILBOX #  *******  161  *******
 010312  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010314  000000                        HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
```

TEST #44 - TEST THAT 'NEW' STATUS IS CORRECT

```
                                        .SBTTL  TEST #44 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;******************************************************************************
                                        ;TEST 44 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;******************************************************************************
010316  032737  000001  177766  TST44:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
010324  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
010326  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
010334  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010336  000000                          HALT                    ;CPU POWER BIT FOUND SET
010340  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
010346  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
010352  022737  000044  000304          CMP     #44,$TESTN      ;SEQUENCE ERROR?
010360  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
010362  000137  010624                  JMP     TST45-12                ;JUMP TO ERROR HALT ON SEQ ERROR
010366                          1000$:
922 010366  012706  000500              MOV     #BUFF,SP
923 010372  012737  010406  000010      MOV     #1$,RTRAP
924 010400  005037  000012              CLR     RTRAP+2 ;CLEAR FUTURE PRIORITY AND CC
925 010404  000100                      JMP     R0
926 010406                          1$:                             ;TEST FOR "C" CLEARED
927 010406  100005                      BPL     2$
010410  012737  000162  000302          MOV     #162,$FATAL     ;MOVE TO MAILBOX #  *******  162  *******
010416  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010420  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
928 010422                          2$:
010422  001005                          BNE     3$
010424  012737  000163  000302          MOV     #163,$FATAL     ;MOVE TO MAILBOX #  *******  163  *******
010432  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010434  000000                          HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
929 010436                          3$:
010436  102005                          BVC     4$
010440  012737  000164  000302          MOV     #164,$FATAL     ;MOVE TO MAILBOX #  *******  164  *******
010446  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010450  000000                          HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
930 010452                          4$:
010452  103005                          BCC     5$
010454  012737  000165  000302          MOV     #165,$FATAL     ;MOVE TO MAILBOX #  *******  165  *******
010462  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010464  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
931 010466  032737  000357  177776  5$:  BIT     #357,CC         ;TEST PRIORITY
932 010474  001405                      BEQ     6$
010476  012737  000166  000302          MOV     #166,$FATAL     ;MOVE TO MAILBOX #  *******  166  *******
010504  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
010506  000000                          HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 726
933 010510  012706  000500          6$:  MOV     #BUFF,SP
934 010514  012737  010532  000010      MOV     #7$,RTRAP
935 010522  012737  000357  000012      MOV     #357,RTRAP+2    ;SET NEW "CC" AND PRIORITY
```

```
936 010530  000100                          JMP     R0              ;TRAP HERE
937 010532                          7$:
    010532  100405                          BMI     8$
    010534  012737  000167  000302          MOV     #167,$FATAL     ;MOVE TO MAILBOX #  *******  167  *******
    010542  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010544  000000                          HALT                    ;N NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 707

938 010546                          8$:
    010546  001405                          BEQ     9$
    010550  012737  000170  000302          MOV     #170,$FATAL     ;MOVE TO MAILBOX #  *******  170  *******
    010556  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010560  000000                          HALT                    ;Z NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 701

939 010562                          9$:
    010562  102405                          BVS     10$
    010564  012737  000171  000302          MOV     #171,$FATAL     ;MOVE TO MAILBOX #  *******  171  *******
    010572  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010574  000000                          HALT                    ;V NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 673

940 010576                          10$:
    010576  103405                          BCS     11$
    010600  012737  000172  000302          MOV     #172,$FATAL     ;MOVE TO MAILBOX #  *******  172  *******
    010606  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010610  000000                          HALT                    ;C NOT SET
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 665

941 010612  013706  177776          11$:    MOV     CC,SP
942 010616  022706  000357                  CMP     #357,SP
943 010622  001405                          BEQ     TST45
    010624  012737  000173  000302          MOV     #173,$FATAL     ;MOVE TO MAILBOX #  *******  173  *******
    010632  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    010634  000000                          HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 653
```

```
944                                     .SBTTL  TEST #45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS
                                        ;******************************************************************
                                        ;TEST 45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS
                                        ;******************************************************************
 010636  032737  000001  177766  TST45: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 010644  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 010646  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 010654  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010656  000000                        HALT                    ;CPU POWER BIT FOUND SET
 010660  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 010666  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 010672  022737  000045  000304        CMP     #45,$TESTN      ;SEQUENCE ERROR?
 010700  001006                        BNE     TST46-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
945 010702  012706  000500            MOV     #BUFF,SP        ;STACK POINTER SETUP
946 010706  012737  010730  000010    MOV     #1$,RTRAP       ;RETURN LOCATION
947 010714  004000                    JSR     R0,R0           ;RESERVED INS, SHOULD TRAP
948 010716  012737  000174  000302    MOV     #174,$FATAL     ;MOVE TO MAILBOX #  *******  174  *******
 010724  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010726  000000                        HALT                    ;DIDN'T TRAP,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 764
949 010730                      1$:
```

```
 950                                    .SBTTL  TEST #46 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;******************************************************************************
                                        ;TEST 46 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;******************************************************************************
 010730  032737  000001  177766  TST46:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 010736  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 010740  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 010746  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 010750  000000                         HALT                    ;CPU POWER BIT FOUND SET
 010752  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 010760  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 010764  022737  000046  000304         CMP     #46,$TESTN      ;SEQUENCE ERROR?
 010772  001011                         BNE     TST47-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
 951 010774  012706  000500             MOV     #BUFF,SP        ;STACK POINTER SETUP
 952 011000  012737  011010  000010     MOV     #1$,RTRAP       ;RETURN POINTER
 953 011006  004000                     JSR     R0,R0           ;RESERVED INS
 954 011010  020627  000474         1$:     CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
 955 011014  001405                     BEQ     TST47
 011016  012737  000175  000302         MOV     #175,$FATAL     ;MOVE TO MAILBOX #  *******  175  *******
 011024  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 011026  000000                         HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 761
```

```
956                                     .SBTTL  TEST #47 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
                                        ;TEST 47 - TEST THAT PROPER P.C. IS SAVED
                                        ;*****************************************************************
    011030  032737  000001  177766  TST47:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
    011036  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
    011040  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
    011046  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011050  000000                          HALT                    ;CPU POWER BIT FOUND SET
    011052  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
    011060  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
    011064  022737  000047  000304          CMP     #47,$TESTN      ;SEQUENCE ERROR?
    011072  001012                          BNE     TST50-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
957 011074  012706  000500                  MOV     #BUFF,SP        ;STACK POINTER SETUP
958 011100  012737  011110  000010          MOV     #1$,RTRAP       ;RETURN FROM TRAP POINTER
959 011106  004000                          JSR     R0,R0           ;TRAP ON THIS INS
960 011110  022737  011110  000474  1$:     CMP     #1$,BUFF-4      ;CHECK FOR INCREMENED P.C.
961 011116  001405                          BEQ     TST50
    011120  012737  000176  000302          MOV     #176,$FATAL     ;MOVE TO MAILBOX #  *******  176  *******
    011126  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011130  000000                          HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                    ;TO SCOPE REPLACE HALT WITH 240
                                                                    ;AND REPLACE NEXT INST WITH 760
962
```

```
                                .SBTTL  TEST #50 - TEST THAT 'OLD'' CC AND PRI ARE PLACED ON STACK
                                ;*****************************************************************
                                ;TEST 50 - TEST THAT 'OLD'' CC AND PRI ARE PLACED ON STACK
                                ;*****************************************************************
011132  032737  000001  177766  TST50:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011140  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
011142  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
011150  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011152  000000                         HALT                    ;CPU POWER BIT FOUND SET
011154  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
011162  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
011166  022737  000050  000304         CMP     #50,$TESTN      ;SEQUENCE ERROR?
011174  001040                         BNE     TST51-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
964 011176  012706  000500             MOV     #BUFF,SP        ;SET UP
965 011202  012737  011220  000010     MOV     #1$,RTRAP       ;SET UP
966 011210  005037  177776             CLR     CC              ;CLEAR CC AND PRICRITY
967 011214  000257                     CCC
968 011216  004000                     JSR     R0,R0           ;TRAP
969 011220  023727  000476  000000  1$:    CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
970 011226  001405                     BEQ     2$
    011230  012737  000177  000302     MOV     #177,$FATAL     ;MOVE TO MAILBOX #  *******  177  *******
    011236  005212                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011240  000000                     HALT                    ;INCORRCT STATUS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 755
971 011242  012706  000500         2$:     MOV     #BUFF,SP        ;SET UP
972 011246  012737  011266  000010     MOV     #3$,RTRAP       ;SET UP
973 011254  012737  000357  177776     MOV     #357,CC         ;SET PRIORITY
974 011262  000277                     SCC                     ;SET CC
975 011264  004000                     JSR     R0,R0           ;TRAP
976 011266  023727  000476  000357  3$:    CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
977 011274  001405                     BEQ     TST51
    011276  012737  000200  000302     MOV     #200,$FATAL     ;MOVE TO MAILBOX #  *******  200  *******
    011304  005212                     INC     (R2)            ;SET MSGTYP TO FATAL ERROR
    011306  000000                     HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 732
```

```
                                        .SBTTL  TEST #51 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*****************************************************************
                                        ;TEST 51 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*****************************************************************
011310  032737  000001  177766 TST51:   BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011316  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
011320  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
011326  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011330  000000                          HALT                    ;CPU POWER BIT FOUND SET
011332  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
011340  005237  000304        100$:     INC     $TESTN          ;UPDATE TEST NUMBER
011344  022737  000051  000304          CMP     #51,$TESTN      ;SEQUENCE ERROR?
011352  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
011354  000137  011614                  JMP     TST52-1<        ;JUMP TO ERROR HALT ON SEQ ERROR
011360                          1000$:
011360  012706  000500                  MOV     #BUFF,SP
011364  012737  011400  000010          MOV     #1$,RTRAP
011372  005037  000012                  CLR     RTRAP+2         ;CLEAR FUTURE PRIORITY AND CC
011376  004000                          JSR     R0,R0
011400                          1$:                             ;TEST FOR "C" CLEARED
011400  100005                          BPL     2$
011402  012737  000201  000302          MOV     #201,$FATAL     ;MOVE TO MAILBOX #  *******  201  *******
011410  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011412  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
011414                          2$:
011414  001005                          BNE     3$
011416  012737  000202  000302          MOV     #202,$FATAL     ;MOVE TO MAILBOX #  *******  202  *******
011424  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011426  000000                          HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 753
011430                          3$:
011430  102005                          BVC     4$
011432  012737  000203  000302          MOV     #203,$FATAL     ;MOVE TO MAILBOX #  *******  203  *******
011440  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011442  000000                          HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 745
011444                          4$:
011444  103005                          BCC     5$
011446  012737  000204  000302          MOV     #204,$FATAL     ;MOVE TO MAILBOX #  *******  204  *******
011454  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011456  000000                          HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 737
011460  013700  177776          5$:     MOV     CC,R0           ;TEMP STORAGE
011464  001405                          BEQ     6$
011466  012737  000205  000302          MOV     #205,$FATAL     ;MOVE TO MAILBOX #  *******  205  *******
011474  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
011476  000000                          HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 727
011500  012706  000500          6$:     MOV     #BUFF,SP
011504  012737  011522  000010          MOV     #7$,RTRAP
011512  012737  000357  000012          MOV     #357,RTRAP+2    ;SET NEW "CC" AND PRIORITY
```

```
  993 011520  004000                        JSR     R0,R0           ;TRAP HERE
  994 011522                          7$:
      011522  100405                        BMI     8$
      011524  012737  000206  000302        MOV     #206,$FATAL     ;MOVE TO MAILBOX #  *******  206  *******
      011532  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      011534  000000                        HALT                    ;N NOT SET
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 710
  995 011536                          8$:
      011536  001405                        BEQ     9$
      011540  012737  000207  000302        MOV     #207,$FATAL     ;MOVE TO MAILBOX #  *******  207  *******
      011546  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      011550  000000                        HALT                    ;Z NOT SET
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 702
  996 011552                          9$:
      011552  102405                        BVS     10$
      011554  012737  000210  000302        MOV     #210,$FATAL     ;MOVE TO MAILBOX #  *******  210  *******
      011562  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      011564  000000                        HALT                    ;V NOT SET
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 674
  997 011566                          10$:
      011566  103405                        BCS     11$
      011570  012737  000211  000302        MOV     #211,$FATAL     ;MOVE TO MAILBOX #  *******  211  *******
      011576  005212                        INC     (R2)             SET MSGTYP TO FATAL ERROR
      011600  000000                        HALT                    ;C NOT SET
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 666
  998 011602  013700  177776          11$:  MOV     CC,R0
  999 011606  022700  000357                MOV     #357,R0
 1000 011612  001405                        BEQ     TST52
      011614  012737  000212  000302        MOV     #212,$FATAL     ;MOVE TO MAILBOX #  *******  212  *******
      011622  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      011624  000000                        HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 654
 1001
```

```
 'U02                                   .SBTTL  TEST #52 - TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS
                                        ;******************************************************************
                                        ;TEST 52 - TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS
                                        ;******************************************************************
      011626  032737  000001  177766  TST52:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
      011634  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
      011636  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
      011644  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      011646  000000                          HALT                    ;CPU POWER BIT FOUND SET
      011650  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
      011656  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
      011662  022737  000052  000304          CMP     #52,$TESTN      ;SEQUENCE ERROR?
      011670  001040                          BNE     3$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1003  011672  005037  177766                  CLR     CPUERR          ;CLEAR CPU ERROR REGISTER
1004  011676  005737  177766                  TST     CPUERR          ;VERIFY THAT IT CLEARED
1005  011702  001405                          BEQ     1$
      011704  012737  000213  000302          MOV     #213,$FATAL     ;MOVE TO MAILBOX #  *******  213  *******
      011712  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      011714  000000                          HALT                    ;CPU ERROR REG FAILED TO CLEAR
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 765
1006  011716  012706  000500          1$:     MOV     #BUFF,SP        ;STACK POINTER SETUP
1007  011722  012737  011746  000004          MOV     #2$,RTRAP5      ;RETURN LOCATION
1008  011730  005737  000001                  TST     1               ;ILLEGAL ADDRESS INS, SHOULD TRAP
1009  011734  012737  000214  000302          MOV     #214,$FATAL     ;MOVE TO MAILBOX #  *******  214  *******
      011742  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      011744  000000                          HALT                    ;ILLEGAL ADDRESS DID NOT TRAP
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 751
1010  011746                            2$:
1011  011746  013737  177766  000502          MOV     CPUERR,RCPUER   ;READ AND SAVE CPU ERROR REGISTER
1012  011754  042737  177413  000502          BIC     #CERMSK,RCPUER  ;MASK OFF UNUSED BITS OF CPU ERROR REG
1013  011762  022737  000100  000502          CMP     #100,RCPUER     ;ODD ADDRESS BIT SET?
1014  011770  001405                          BEQ     4$
      011772                            3$:
      011772  012737  000215  000302          MOV     #215,$FATAL     ;MOVE TO MAILBOX #  *******  215  *******
      012000  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      012002  000000                          HALT                    ;INCORRECT CPU ERROR REG CONTENTS, OR WRONG $TSTNM
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 732
1015  012004  005037  177766          4$:     CLR     CPUERR          ;CLEAR ODD ADDRESS BIT
1016
```

```
1017                                    .SBTTL  TEST #53 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*************************************************************************
                                        ;TEST 53 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
                                        ;*************************************************************************
 012010  032737  000001  177766 TST53:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 012016  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 012020  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 012026  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 012030  000000                         HALT                    ;CPU POWER BIT FOUND SET
 012032  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 012040  005237  000304         100$:  INC     $TESTN          ;UPDATE TEST NUMBER
 012044  022737  000053  000304         CMP     #53,$TESTN      ;SEQUENCE ERROR?
 012052  001012                         BNE     TST54-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1018 012054  012706  000500             MOV     #BUFF,SP        ;STACK POINTER SETUP
1019 012060  012737  012072  000004     MOV     #1$,RTRAP5      ;RETURN POINTER
1020 012066  005737  000001             TST     1               ;RESERVED INS
1021 012072  020627  000474      1$:    CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
1022 012076  001405                     BEQ     TST54
 012100  012737  000216  000302         MOV     #216,$FATAL     ;MOVE TO MAILBOX #  *******  216  *******
 012106  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 012110  000000                         HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 760
```

```
1023                                    .SBTTL  TEST #54 - TEST THAT PROPER P.C. IS SAVED
                                        ;*************************************************************************
                                        ;TEST 54 - TEST THAT PROPER P.C. IS SAVED
                                        ;*************************************************************************
 012112  032737  000001  177766 TST54:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 012120  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 012122  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 012130  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 012132  000000                         HALT                    ;CPU POWER BIT FOUND SET
 012134  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 012142  005237  000304         100$:  INC     $TESTN          ;UPDATE TEST NUMBER
 012146  022737  000054  000304         CMP     #54,$TESTN      ;SEQUENCE ERROR?
 012154  001013                         BNE     TST55-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1024 012156  012706  000500             MOV     #BUFF,SP        ;STACK POINTER SETUP
1025 012162  012737  012174  000004     MOV     #1$,RTRAP5      ;RETURN FROM TRAP POINTER
1026 012170  005737  000001             TST     1               ;TRAP ON THIS INSTRUCTION
1027 012174  022737  012174  000474 1$:  CMP     #1$,BUFF-4      ;CHECK FOR INCREMENTED P.C.
1028 012202  001405                     BEQ     TST55
 012204  012737  000217  000302         MOV     #217,$FATAL     ;MOVE TO MAILBOX #  *******  217  *******
 012212  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 012214  000000                         HALT                    ;INCORRECT P.C.,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 757
```

```
1029                                    .SBTTL  TEST #55 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
                                        ;TEST 55 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
                                        ;*****************************************************************
     012216  032737  000001  177766  TST55:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     012224  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     012226  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     012234  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012236  000000                          HALT                    ;CPU POWER BIT FOUND SET
     012240  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     012246  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     012252  022737  000055  000304          CMP     #55,$TESTN      ;SEQUENCE ERROR?
     012260  001050                          BNE     TST56-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1030 012262  000405                          BR      90$             ;BRANCH OVER UNUSED ERROR CODE
1031 012264  012737  000220  000302          MOV     #220,$FATAL     ;MOVE TO MAILBOX #  *******  220  *******
     012272  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012274  000000                          HALT                    ;ERROR 177 RESERVED FOR CPU POWER BIT FOUND SET
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 771
1032 012276  012706  000500          90$:    MOV     #BUFF,SP        ;SET UP
1033 012302  012737  012322  000004          MOV     #1$,RTRAP5      ;SET UP
1034 012310  005037  177776                  CLR     CC              ;CLEAR CC AND PRIORITY
1035 012314  000257                          CCC
1036 012316  005737  000001                  TST     1               ;TRAP
1037 012322  023727  000476  000000  1$:     CMP     BUFF-2,#0       ;TEST THAT OLD STATUS WENT TO STACK
1038 012330  001405                          BEQ     2$
     012332  012737  000221  000302          MOV     #221,$FATAL     ;MOVE TO MAILBOX #  *******  221  *******
     012340  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012342  000000                          HALT                    ;INCORRECT STATUS
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 746
1039 012344  012706  000500          2$:     MOV     #BUFF,SP        ;SET UP
1040 012350  012737  012372  000004          MOV     #3$,RTRAP5      ;SET UP
1041 012356  012737  000357  177776          MOV     #357,CC         ;SET PRIORITY
1042 012364  000277                          SCC                     ;SET CC
1043 012366  005737  000001                  TST     1               ;TRAP
1044 012372  023727  000476  000357  3$:     CMP     BUFF-2,#357     ;COMPARES STATUS ON STACK
1045 012400  001405                          BEQ     TST56
     012402  012737  000222  000302          MOV     #222,$FATAL     ;MOVE TO MAILBOX #  *******  222  *******
     012410  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012412  000000                          HALT                    ;INCORRCT STATUS ON STACK,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 722
```

```
1046                                    .SBTTL  TEST #56 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*************************************************************
                                        ;TEST 56 - TEST THAT 'NEW' STATUS IS CORRECT
                                        ;*************************************************************
012414  032737  000001  177766  TST56:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
012422  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
012424  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
012432  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
012434  000000                          HALT                    ;CPU POWER BIT FOUND SET
012436  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
012444  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
012450  022737  000056  000304          CMP     #56,$TESTN      ;SEQUENCE ERROR?
012456  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
012460  000137  012726                  JMP     TST57-12                ;JUMP TO ERROR HALT ON SEQ ERROR
012464                          1000$:
1047 012464  012706  000500            MOV     #BUFF,SP
1048 012470  012737  012506  000004    MOV     #1$,RTRAP5
1049 012476  005037  000006            CLR     RTRAP5+2        ;CLEAR FUTURE PRIORITY AND CC
1050 012502  005737  000001            TST     1               ;TRAP HERE
1051 012506                     1$:            ;TEST FOR "C" CLEARED
1052 012506  100005                    BPL     2$
     012510  012737  000223  000302    MOV     #223,$FATAL     ;MOVE TO MAILBOX #  *******  223  *******
     012516  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012520  000000                    HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 760
1053 012522                     2$:
     012522  001005                    BNE     3$
     012524  012737  000224  000302    MOV     #224,$FATAL     ;MOVE TO MAILBOX #  *******  224  *******
     012532  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012534  000000                    HALT                    ;Z NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 752
1054 012536                     3$:
     012536  102005                    BVC     4$
     012540  012737  000225  000302    MOV     #225,$FATAL     ;MOVE TO MAILBOX #  *******  225  *******
     012546  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012550  000000                    HALT                    ;V NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 744
1055 012552                     4$:
     012552  103005                    BCC     5$
     012554  012737  000226  000302    MOV     #226,$FATAL     ;MOVE TO MAILBOX #  *******  226  *******
     012562  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012564  000000                    HALT                    ;C NOT CLEARED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 736
1056 012566  032737  000357  177776  5$:     BIT     #357,CC         ;TEST PRIORITY FOR ZERO
1057 012574  001405                    BEQ     6$
     012576  012737  000227  000302    MOV     #227,$FATAL     ;MOVE TO MAILBOX #  *******  227  *******
     012604  005212                    INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012606  000000                    HALT                    ;PRIORITY NOT ZERO
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 725
1058 012610  012706  000500            6$:     MOV     #BUFF,SP
1059 012614  012737  012634  000004    MOV     #7$,RTRAP5
1060 012622  012737  000357  000006    MOV     #357,RTRAP5+2   ;SET NEW "CC" AND PRIORITY
```

TEST #56 - TEST THAT 'NEW' STATUS IS CORRECT

```
1061 012630  005737  000001                    TST     1               ;TRACE HERE
1062 012634                              7$:
     012634  100405                             BMI     8$
     012636  012737  000230  000302             MOV     #230,$FATAL     ;MOVE TO MAILBOX #  *******  230  *******
     012644  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012646  000000                             HALT                    ;N NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 705

1063 012650                              8$:
     012650  001405                             BEQ     9$
     012652  012737  000231  000302             MOV     #231,$FATAL     ;MOVE TO MAILBOX #  *******  231  *******
     012660  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012662  000000                             HALT                    ;Z NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 677

1064 012664                              9$:
     012664  102405                             BVS     10$
     012666  012737  000232  000302             MOV     #232,$FATAL     ;MOVE TO MAILBOX #  *******  232  *******
     012674  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012676  000000                             HALT                    ;V NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 671

1065 012700                              10$:
     012700  103405                             BCS     11$
     012702  012737  000233  000302             MOV     #233,$FATAL     ;MOVE TO MAILBOX #  *******  233  *******
     012710  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012712  000000                             HALT                    ;C NOT SET
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 663

1066 012714  013700  177776              11$:   MOV     CC,R0
1067 012720  022700  000357                     CMP     #357,R0
1068 012724  001405                             BEQ     TST57
     012726  012737  000234  000302             MOV     #234,$FATAL     ;MOVE TO MAILBOX #  *******  234  *******
     012734  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012736  000000                             HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 651
```

```
1069                                    .SBTTL  TEST #57 - TEST THAT DEC R6 TO A VALUE LESS  400 TRAPS
                                        ;*****************************************************************************
                                        ;TEST 57 - TEST THAT DEC R6 TO A VALUE LESS  400 TRAPS
                                        ;*****************************************************************************
     012740  032737  000001  177766     TST57:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     012746  001410                             BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     012750  012737  000177  000302             MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     012756  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     012760  000000                             HALT                    ;CPU POWER BIT FOUND SET
     012762  042737  000001  177766             BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     012770  005237  000304             100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     012774  022737  000057  000304             CMP     #57,$TESTN      ;SEQUENCE ERROR?
     013002  001027                             BNE     2$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1070 013004  005037  177766                     CLR     CPUERR          ;CLEAR CPU ERROR REGISTER
1071 013010  012706  000150                     MOV     #150,R6         ;R6 = 150
1072 013014  012737  013036  000004             MOV     #1$,4           ;STACK OVERFLOW TRAP POINTER
1073 013022  005746                             TST     -(R6)           ;WITH R6 = 150 SHOULD TRAP
1074 013024  012737  000235  000302             MOV     #235,$FATAL     ;MOVE TO MAILBOX #  *******  235  *******
     013032  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     013034  000000                             HALT                    ;SHOULD HAVE TRAPPED,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 762
1075 013036                             1$:
1076 013036  013737  177766  000502             MOV     CPUERR,RCPUER   ;SAVE CPU ERROR REGISTER
1077 013044  042737  177413  000502             BIC     #CERMSK,RCPUER  ;MASK OFF UNUSED CPU ERROR REG BITS
1078 013052  022737  000004  000502             CMP     #4,RCPUER       ;IS YELLOW ZONE BIT SET?
1079 013060  001405                             BEQ     3$
     013062                             2$:
     013062  012737  000236  000302             MOV     #236,$FATAL     ;MOVE TO MAILBOX #  *******  236  *******
     013070  005212                             INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     013072  000000                             HALT                    ;INCORRECT CPU ERROR REGISTER CONTENTS, OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 743
1080 013074  005037  177766             3$:     CLR     CPUERR          ;CLEAR YELLOW ZONE BIT
```

```
1081                                    .SBTTL  TEST #60 - TEST FOR DEC OF R6 ON OVERFLOW TRAP
                                        ;******************************************************************************
                                        ;TEST 60 - TEST FOR DEC OF R6 ON OVERFLOW TRAP
                                        ;******************************************************************************
013100  032737  000001  177766  TST60:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013106  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013110  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013116  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013120  000000                          HALT                    ;CPU POWER BIT FOUND SET
013122  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013130  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013134  022737  000060  000304          CMP     #60,$TESTN      ;SEQUENCE ERROR?
013142  001011                          BNE     TST61-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1082 013144  012706  000150            MOV     #150,R6         ;R6 = 150
1083 013150  012737  013160  000004    MOV     #1$,4           ;TRAP POINTER
1084 013156  005746                    TST     -(R6)           ;WITH R6 = 150 SHOULD TRAP
1085 013160  020627  000142    1$:     CMP     R6,#142         ;DID R6 DECREMENT
1086 013164  001405                    BEQ     TST61
013166  012737  000237  000302          MOV     #237,$FATAL     ;MOVE TO MAILBOX #  *******  237  *******
013174  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013176  000000                          HALT                    ;R6 NOT = 142,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
```

```
1087                                    .SBTTL  TEST #61 - TEST DIFFERENT TYPES OF OVERFLOW
                                        ;*****************************************************************
                                        ;TEST 61 - TEST DIFFERENT TYPES OF OVERFLOW
                                        ;*****************************************************************
013200  032737  000001  177766  TST61:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013206  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013210  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013216  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013220  000000                          HALT                    ;CPU POWER BIT FOUND SET
013222  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013230  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013234  022737  000061  000304          CMP     #61,$TESTN      ;SEQUENCE ERROR?
013242  001043                          BNE     TST62-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1088 013244  012706  000150            MOV     #150,R6
1089 013250  005037  000146            CLR     146             ;STATUS WORD OF LOC 10
1090 013254  012737  013264  000004    MOV     #1$,4           ;RETURN TO LOC 4
1091 013262  005246                    INC     -(6)
1092 013264  005737  000146    1$:     TST     146
1093 013270  001005                    BNE     2$
013272  012737  000240  000302          MOV     #240,$FATAL     ;MOVE TO MAILBOX #  *******  240  *******
013300  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013302  000000                          HALT                    ;INCREMENT OPERATION NOT INHIBITED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 757
1094 013304  012705  001000    2$:     MOV     #1000,R5
1095 013310  012706  000400            MOV     #400,R6
1096 013314  012737  013336  000004    MOV     #3$,4
1097 013322  124645                    CMPB    -(R6),-(R5)
1098 013324  012737  000241  000302    MOV     #241,$FATAL     ;MOVE TO MAILBOX #  *******  241  *******
013332  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013334  000000                          HALT                    ;STACK = 400 AND DECREMENTED, SHOULD TRAP
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 742
1099 013336  012706  000400    3$:     MOV     #400,R6
1100 013342  012737  013364  000004    MOV     #5$,4
1101 013350  134546                    BITB    -(R5),-(R6)
1102 013352                    4$:
013352  012737  000242  000302          MOV     #242,$FATAL     ;MOVE TO MAILBOX #  *******  242  *******
013360  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013362  000000                          HALT                    ;NO STACK OVERFLOW,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 727
1103 013364                    5$:
1113
```

```
1114                                    .SBTTL  TEST #62 - TEST THAT AN 10 CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
                                        ;TEST 62 - TEST THAT AN 10 CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
013364  032737  000001  177766  TST62: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013372  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013374  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013402  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013404  000000                        HALT                    ;CPU POWER BIT FOUND SET
013406  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013414  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013420  022737  000062  000304        CMP     #62,$TESTN      ;SEQUENCE ERROR?
013426  001011                        BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
013430  012706  000400                MOV     #400,R6         ;SET UP STACK TO OVERFLOW
013434  012737  013452  000010        MOV     #1$,10  ;SET UP 10 VECTOR
013442  012737  013464  000004        MOV     #2$,4           ;SET UP OVERFLOW VECTOR
013450  000010                        10                      ;THIS TRAP SHOULD CAUSE OVERFLOW
013452                          1$:
013452  012737  000243  000302        MOV     #243,$FATAL     ;MOVE TO MAILBOX #  *******  243  *******
013460  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013462  000000                        HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
013464  012737  000012  000010  2$:    MOV     #10+2,10
```

```
1115                                    .SBTTL  TEST #63 - TEST THAT AN IOT CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
                                        ;TEST 63 - TEST THAT AN IOT CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
013472  032737  000001  177766  TST63:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013500  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013502  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013510  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013512  000000                          HALT                    ;CPU POWER BIT FOUND SET
013514  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013522  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013526  022737  000063  000304          CMP     #63,$TESTN      ;SEQUENCE ERROR?
013534  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
013536  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
013542  012737  013560  000020          MOV     #1$,20  ;SET UP IOT VECTOR
013550  012737  013572  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
013556  000004                          IOT                     ;THIS TRAP SHOULD CAUSE OVERFLOW
013560                          1$:
013560  012737  000244  000302          MOV     #244,$FATAL     ;MOVE TO MAILBOX #  *******  244  *******
013566  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013570  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
013572  012737  000022  000020  2$:     MOV     #20+2,20
```

```
1116                                    .SBTTL  TEST #64 - TEST THAT AN EMT CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************************
                                        ;TEST 64 - TEST THAT AN EMT CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************************
013600  032737  000001  177766  TST64:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013606  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013610  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013616  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013620  000000                          HALT                    ;CPU POWER BIT FOUND SET
013622  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013630  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013634  022737  000064  000304          CMP     #64,$TESTN      ;SEQUENCE ERROR?
013642  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
013644  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
013650  012737  013666  000030          MOV     #1$,30  ;SET UP EMT VECTOR
013656  012737  013700  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
013664  104000                          EMT                     ;THIS TRAP SHOULD CAUSE OVERFLOW
013666                          1$:
013666  012737  000245  000302          MOV     #245,$FATAL     ;MOVE TO MAILBOX #  *******  245  *******
013674  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013676  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761

013700  012737  000032  000030  2$:     MOV     #30+2,30
```

```
1117                                    .SBTTL  TEST #65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************************
                                        ;TEST 65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************************
013706  032737  000001  177766  TST65:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013714  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013716  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013724  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
013726  000000                          HALT                    ;CPU POWER BIT FOUND SET
013730  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
013736  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
013742  022737  000065  000304          CMP     #65,$TESTN      ;SEQUENCE ERROR?
013750  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
013752  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
013756  012737  013774  000034          MOV     #1$,34  ;SET UP TRAP VECTOR
013764  012737  014006  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
013772  104400                          TRAP                    ;THIS TRAP SHOULD CAUSE OVERFLOW
013774                          1$:
013774  012737  000246  000302          MOV     #246,$FATAL     ;MOVE TO MAILBOX #  *******  246  *******
014002  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014004  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761

014006  012737  000036  000034  2$:     MOV     #34+2,34
```

```
1118                                    .SBTTL  TEST #66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
                                        ;TEST 66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP
                                        ;*****************************************************************
014014  032737  000001  177766  TST66:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014022  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
014024  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
014032  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014034  000000                          HALT                    ;CPU POWER BIT FOUND SET
014036  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
014044  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
014050  022737  000066  000304          CMP     #66,$TESTN      ;SEQUENCE ERROR?
014056  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
014060  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
014064  012737  014102  000014          MOV     #1$,14  ;SET UP TRT VECTOR
014072  012737  014114  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
014100  000003                          TRT                     ;THIS TRAP SHOULD CAUSE OVERFLOW
014102                          1$:
014102  012737  000247  000302          MOV     #247,$FATAL     ;MOVE TO MAILBOX #  *******  247  *******
014110  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014112  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761

014114  012737  000016  000014  2$:     MOV     #14+2,14
```

```
 1119                                   .SBTTL  TEST #67 - TEST THAT AN ILLA CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************
                                        ;TEST 67 - TEST THAT AN ILLA CAUSES AN OVERFLOW TRAP
                                        ;******************************************************************
 014122  032737  000001  177766  TST67: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 014130  001410                        BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 014132  012737  000177  000302        MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 014140  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 014142  000000                        HALT                    ;CPU POWER BIT FOUND SET
 014144  042737  000001  177766        BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 014152  005237  000304        100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 014156  022737  000067  000304        CMP     #67,$TESTN      ;SEQUENCE ERROR?
 014164  001011                        BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
 014166  012706  000400                MOV     #400,R6         ;SET UP STACK TO OVERFLOW
 014172  012737  014210  000010        MOV     #1$,10  ;SET UP ILLA VECTOR
 014200  012737  014222  000004        MOV     #2$,4           ;SET UP OVERFLOW VECTOR
 014206  004700                        ILLA                    ;THIS TRAP SHOULD CAUSE OVERFLOW
 014210                         1$:
 014210  012737  000250  000302        MOV     #250,$FATAL     ;MOVE TO MAILBOX #  *******  250  *******
 014216  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 014220  000000                        HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761

 014222  012737  000012  000010  2$:   MOV     #10+2,10
 1120 014230  020627  000370            CMP     R6,#370         ;STACK PUSHED FOUR WORDS?
 1121 014234  001405                    BEQ     TST70
 014236  012737  000251  000302        MOV     #251,$FATAL     ;MOVE TO MAILBOX #  *******  251  *******
 014244  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 014246  000000                        HALT                    ;CORRECT # (4) OF WORDS WERE NOT PUSHED ONTO STACK
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 746
```

```
1122                                    .SBTTL  TEST #70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP
                                        ;*************************************************************************
                                        ;TEST 70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP
                                        ;*************************************************************************
014250  032737  000001  177766  TST70:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014256  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
014260  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
014266  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014270  000000                          HALT                    ;CPU POWER BIT FOUND SET
014272  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
014300  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
014304  022737  000070  000304          CMP     #70,$TESTN      ;SEQUENCE ERROR?
014312  001011                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
014314  012706  000400                  MOV     #400,R6         ;SET UP STACK TO OVERFLOW
014320  012737  014336  000010          MOV     #1$,10  ;SET UP ILLB VECTOR
014326  012737  014350  000004          MOV     #2$,4           ;SET UP OVERFLOW VECTOR
014334  000100                          ILLB                    ;THIS TRAP SHOULD CAUSE OVERFLOW
014336                          1$:
014336  012737  000252  000302          MOV     #252,$FATAL     ;MOVE TO MAILBOX #  *******  252  *******
014344  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014346  000000                          HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 761
014350  012737  000012  000010  2$:     MOV     #10+2,10
1123
```

```
1124                                    .SBTTL  TEST #71 - TEST FOR FALSE OVERFLOW TRAP
                                        ;******************************************************************************
                                        ;TEST 71 - TEST FOR FALSE OVERFLOW TRAP
                                        ;******************************************************************************
     014356  032737  000001  177766  TST71:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     014364  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     014366  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     014374  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     014376  000000                          HALT                    ;CPU POWER BIT FOUND SET
     014400  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     014406  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     014412  022737  000071  000304          CMP     #71,$TESTN      ;SEQUENCE ERROR?
     014420  001023                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1125
1126 014422  012737  014470  000004          MOV     #1$,4           ;SET UP OVERFLOW POINTER
1127 014430  012706  001002                  MOV     #1002,R6
1128 014434  005746                          TST     -(R6)           ;SHOULD NOT OVERFLOW
1129 014436  012706  002002                  MOV     #2002,R6
1130 014442  005746                          TST     -(R6)           ;SHOULD NOT OVERFLOW
1131 014444  012706  004002                  MOV     #4002,R6
1132 014450  005746                          TST     -(R6)           ;SHOULD NOT OVERFLOW
1133 014452  012706  010002                  MOV     #10002,R6
1134 014456  005746                          TST     -(R6)
1135 014460  012706  020000                  MOV     #20000,R6       ;SHOULD NOT OVERFLOW
1136 014464  005746                          TST     -(R6)
1137 014466  000405                          BR      2$
     014470                       1$:
     014470  012737  000253  000302          MOV     #253,$FATAL     ;MOVE TO MAILBOX #  *******  253  *******
     014476  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     014500  000000                          HALT                    ;STACK OVERFLOWED,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 747
1138 014502  012737  000006  000004  2$:     MOV     #6,4
1139 014510  005037  000006                  CLR     6
```

```
1140                                    .SBTTL  TEST #72 - TEST THAT BIT 4 PSW WILL CAUSE A TRAP TO 14
                                        ;*************************************************************************
                                        ;TEST 72 - TEST THAT BIT 4 PSW WILL CAUSE A TRAP TO 14
                                        ;*************************************************************************
     014514  032737  000001  177766  TST72:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     014522  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     014524  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     014532  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     014534  000000                          HALT                    ;CPU POWER BIT FOUND SET
     014536  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     014544  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     014550  022737  000072  000304          CMP     #72,$TESTN      ;SEQUENCE ERROR?
     014556  001013                          BNE     TST73-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1141 014560  012706  000500                  MOV     #BUFF,SP
1142 014564  012737  014620  000014          MOV     #2$,RTRAP4      ;SET UP TO TRAP TO 14
1143 014572  012746  000020                  MOV     #20,-(SP)       ;PUSH T BIT
1144 014576  012746  014604                  MOV     #1$,-(SP)       ;PUSH PC
1145 014602  000002                          RTI                     ;SET T BIT
1146 014604  000240                  1$:     NOP                     ;TRAP HERE
1147 014606  012737  0002?4  000302          MOV     #254,$FATAL     ;MOVE TO MAILBOX #  *******  254  *******
     014614  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     014616  000000                          HALT                    ;TRACE BIT DID NOT TRAP!,OR WRONG $TESTN
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 757
1148 014620                          2$:
```

```
1149                                    .SBTTL  TEST #73 - TEST  STACK POINTER DECREMENTS
                                        ;******************************************************************************
                                        ;TEST 73 - TEST  STACK POINTER DECREMENTS
                                        ;******************************************************************************
014620  032737  000001  177766  TST73:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014626  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
014630  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
014636  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014640  000000                          HALT                    ;CPU POWER BIT FOUND SET
014642  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
014650  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
014654  022737  000073  000304          CMP     #73,$TESTN      ;SEQUENCE ERROR?
014662  001023                          BNE     TST74-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1150 014664  012706  000500            MOV     #BUFF,SP
1151 014670  012737  014724  000014    MOV     #2$,RTRAP4
1152 014676  012746  000020            MOV     #20,-(SP)       ;PUSH T BIT
1153 014702  012746  014710            MOV     #1$,-(SP)       ;PUSH PC
1154 014706  000002                    RTI                     ;SET T BIT
1155 014710  000240            1$:     NOP                     ;TRAP HERE
1156 014712  012737  000255  000302    MOV     #255,$FATAL     ;MOVE TO MAILBOX #  *******  255  *******
014720  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014722  000000                          HALT                    ;TRACE BIT DID NOT TRAP!
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 757
1157 014724  020627  000474    2$:     CMP     SP,#BUFF-4
1158 014730  001405                    BEQ     TST74
014732  012737  000256  000302          MOV     #256,$FATAL     ;MOVE TO MAILBOX #  *******  256  *******
014740  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
014742  000000                          HALT                    ;STACK POINTER WAS NOT PUSHED BY TRAP,OR WRONG $TESTN
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 747
```

TEST #74 - TEST FOR PROPER PC ON STACK

```
1159                                    .SBTTL  TEST #74 - TEST  FOR PROPER PC ON STACK
                                        ;******************************************************************
                                        ;TEST 74 - TEST  FOR PROPER PC ON STACK
                                        ;******************************************************************
     014744  032737  000001  177766  TST74:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     014752  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     014754  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     014762  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     014764  000000                          HALT                    ;CPU POWER BIT FOUND SET
     014766  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     014774  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     015000  022737  000074  000304          CMP     #74,$TESTN      ;SEQUENCE ERROR?
     015006  001016                          BNE     TST75-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1160 015010  012706  000500                  MOV     #BUFF,SP
1161 015014  012737  015056  000014          MOV     #2$,RTRAP4
1162 015022  012746  000020                  MOV     #20,-(SP)       ;PUSH T BIT
1163 015026  012746  015034                  MOV     #1$,-(SP)       ;PUSH PC
1164 015032  000002                          RTI                     ;SET T BIT
1165 015034  022737  015034  000474  1$:     CMP     #1$,BUFF-4      ;TRAP HERE - SEE IF PC IS CORRECT
1166 015042  001405                          BEQ     TST75
     015044  012737  000257  000300          MOV     #257,$FATAL     ;MOVE TO MAILBOX #  *******  257  *******
     015052  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015054  000000                          HALT                    ;CORRECT PC WAS NOT SAVED ON STACK,OR WRONG $TESTN
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 754

1167 015056                             2$:
```

TEST #75 - TEST THAT RTT POPS T- BIT

```
1168                                    .SBTTL  TEST #75 - TEST  THAT RTT POPS T- BIT
                                        ;*****************************************************************************
                                        ;TEST 75 - TEST  THAT RTT POPS T- BIT
                                        ;*****************************************************************************
     015056  032737  000001  177766  TST75:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     015064  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     015066  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     015074  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015076  000000                          HALT                    ;CPU POWER BIT FOUND SET
     015100  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     015106  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     015112  022737  000075  000304          CMP     #75,$TESTN      ;SEQUENCE ERROR?
     015120  001015                          BNE     TST76-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1169 015122  012706  000500                  MOV     #BUFF,SP
1170 015126  005001                          CLR     R1              ;CLEAR R1
1171 015130  012746  000020                  MOV     #20,-(SP)
1172 015134  012746  015150                  MOV     #1$,-(SP)
1173 015140  012737  015166  000014          MOV     #2$,14
1174 015146  000006                          RTT
1175 015150  000240                  1$:     NOP
1176 015152  001405                          BEQ     TST76
     015154  012737  000260  000302          MOV     #260,$FATAL     ;MOVE TO MAILBOX #  *******  260  *******
     015162  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015164  000000                          HALT                    ;T-BIT DID NOT TRAP,OR WRONG $TESTN
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 755
1177 015166                          2$:
```

```
                                        .SBTTL  TEST #76 - TEST  THAT RTT ALLOWS ONE INST. BEFORE TRAP
                                        ;*************************************************************************
                                        ;TEST 76 - TEST  THAT RTT ALLOWS ONE INST. BEFORE TRAP
                                        ;*************************************************************************
 015166  032737  000001  177766  TST76:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 015174  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 015176  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 015204  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 015206  000000                          HALT                    ;CPU POWER BIT FOUND SET
 015210  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 015216  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 015222  022737  000076  000304          CMP     #76,$TESTN      ;SEQUENCE ERROR?
 015230  001031                          BNE     TST77-12                ;BRANCH TO ERROR HALT ON SEQ ERROR
1179 015232  012705  177777              MOV     #177777,R5
1180 015236  012706  000500          1$:     MOV     #BUFF,SP
1181 015242  012746  000020              MOV     #20,-(SP)
1182 015246  012746  015264              MOV     #2$,-(SP)
1183 015252  012737  015304  000014      MOV     #3$,14
1184 015260  005001                      CLR     R1              ;CLEAR R0
1185 015262  000006                      RTT                     ;SET T-BIT
1186 015264  005201          2$:     INC     R1
1187 015266  005205                      INC     R5
1188 015270  001762                      BEQ     1$              ;DO THIS TEST NO MORE THAN 2 TIMES
1189 015272  012737  000261  000302      MOV     #261,$FATAL     ;MOVE TO MAILBOX #  *******  261  *******
 015300  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 015302  000000                          HALT                    ;DID NOT TRAP
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 752
1190 015304  005301          3$:     DEC     R1              ;SEE IF RTT ALLOWS 1 INST.
1191 015306  001407                      BEQ     4$
1192 015310  005205                      INC     R5              ;DO THIS TEST NO MORE THAN TWO TIMES
1193 015312  001751                      BEQ     1$
 015314  012737  000262  000302          MOV     #262,$FATAL     ;MOVE TO MAILBOX #  *******  262  *******
 015322  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 015324  000000                          HALT                    ;RTT DID NOT ALLOW 1 INST.,OR WRONG $TESTN
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 741
1194 015326                          4$:
```

TEST #77 - TEST THAT RTI DOES NOT ALLOW 1 INST.

```
1195                                    .SBTTL  TEST #77 - TEST  THAT RTI DOES NOT ALLOW 1 INST.
                                        ;*************************************************************************
                                        ;TEST 77 - TEST  THAT RTI DOES NOT ALLOW 1 INST.
                                        ;*************************************************************************
     015326  032737  000001  177766  TST77:  BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     015334  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     015336  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     015344  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015346  000000                          HALT                    ;CPU POWER BIT FOUND SET
     015350  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     015356  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     015362  022737  000077  000304          CMP     #77,$TESTN      ;SEQUENCE ERROR?
     015370  001023                          BNE     TST100-12               ;BRANCH TO ERROR HALT ON SEQ ERROR
1196 015372  012706  000500                  MOV     #BUFF,SP
1197 015376  012746  000020                  MOV     #20,-(SP)
1198 015402  012746  015420                  MOV     #1$,-(SP)
1199 015406  012737  015434  000014          MOV     #2$,14
1200 015414  005001                          CLR     R1
1201 015416  000002                          RTI                     ;SET T-BIT
1202 015420  005201                  1$:     INC     R1              ;RTI SHOULD NOT ALLOW THIS
1203 015422  012737  000263  000302          MOV     #263,$FATAL     ;MOVE TO MAILBOX #  *******  263  *******
     015430  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015432  000000                          HALT                    ;T- BIT DID NOT CAUSE TRAP
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 756
1204 015434  005701                  2$:     TST     R1
1205                                                                 ;RTI SHOULD NOT ALLOW 1 INST. BEFORE TRAP
1206 015436  001405                          BEQ     TST100
     015440  012737  000264  000302          MOV     #264,$FATAL     ;MOVE TO MAILBOX #  *******  264  *******
     015446  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015450  000000                          HALT                    ;RTI DID ALLOW 1 INST. BEFORE TRAP,OR WRONG $TESTN
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 747
```

```
1207                                    .SBTTL  TEST #100 - DOES THE PROCESSOR TRAP WHEN R7 IS ODD?
                                        ;*************************************************************************
                                        ;TEST 100 - DOES THE PROCESSOR TRAP WHEN R7 IS ODD?
                                        ;*************************************************************************
     015452  032737  000001  177766  TST100: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     015460  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     015462  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     015470  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015472  000000                          HALT                    ;CPU POWER BIT FOUND SET
     015474  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     015502  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     015506  022737  000100  000304          CMP     #100,$TESTN     ;SEQUENCE ERROR?
     015514  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
     015516  000137  015762                  JMP     TST101-12               ;JUMP TO ERROR HALT ON SEQ ERROR
     015522                          1000$:
1208 015522  012706  000500                  MOV     #BUFF,R6        ;SET UP STACK POINTER
1209 015526  012737  015552  000004          MOV     #1$,4           ;RETURN FROM TRAP
1210 015534  012707  000001                  MOV     #1,R7           ;PC EQUALS ONE
1211 015540  012737  000265  000302          MOV     #265,$FATAL     ;MOVE TO MAILBOX #  *******  265  *******
     015546  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015550  000000                          HALT                    ;ODD ADDRESS SHOULD HAVE TRAPPED
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 763
1212 015552  022737  000001  000474  1$:     CMP     #1,BUFF-4
1213 015560  001405                          BEQ     2$
     015562  012737  000266  000302          MOV     #266,$FATAL     ;MOVE TO MAILBOX #  *******  266  *******
     015570  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015572  000000                          HALT                    ;CORRECT PC WAS NOT SAVED ON STACK
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 752
1214 015574  012706  000500          2$:     MOV     #BUFF,R6        ;STACK POINTER
1215 015600  012737  015622  000004          MOV     #4$,4
1216 015606  005207                          INC     R7              ;PC BECOMES ODD
1217 015610                          3$:
     015610  012737  000267  000302          MOV     #267,$FATAL     ;MOVE TO MAILBOX #  *******  267  *******
     015616  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015620  000000                          HALT                    ;
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 737
1218 015622  022737  015611  000474  4$:     CMP     #3$+1,BUFF-4
1219 015630  001405                          BEQ     5$
     015632  012737  000270  000302          MOV     #270,$FATAL     ;MOVE TO MAILBOX #  *******  270  *******
     015640  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015642  000000                          HALT                    ;CORRECT PC NOT ON STACK
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 726
1220 015644  012706  000500          5$:     MOV     #BUFF,R6
1221 015650  012737  015672  000004          MOV     #7$,4
1222 015656  005307                  6$:     DEC     R7              ;MAKE PC ODD
1223 015660  012737  000271  000302          MOV     #271,$FATAL     ;MOVE TO MAILBOX #  *******  271  *******
     015666  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     015670  000000                          HALT                    ;SHOULD TRAP
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 713
1224 015672  022737  015657  000474  7$:     CMP     #6$+1,BUFF-4    ;CHECK VALUE OF PC ON STACK
1225 015700  001405                          BEQ     8$
     015702  012737  000272  000302          MOV     #272,$FATAL     ;MOVE TO MAILBOX #  *******  272  *******
```

'EST #100 - DOES THE PROCESSOR TRAP WHEN R7 IS ODD?

```
        015710  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        015712  000000                          HALT                    ;WRONG VALUE ON STACK
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 702
  1226  015714  012706  000500          8$:     MOV     #BUFF,R6
  1227  015720  012737  015744  000004        MOV     #10$,4
  1228  015726  000261                        SEC                     ;CARRY EQUALS A 1
  1229  015730  006107                        ROL     R7              ;PC BECOMES ODD
  1230  015732                          9$:
        015732  012737  000273  000302        MOV     #273,$FATAL     ;MOVE TO MAILBOX #  *******  273  *******
        015740  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        015742  000000                        HALT                    ;ODD ADDRESS DIDN'T TRAP
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 666
  1231  015744  012737  000006  000004  10$:  MOV     #6,4            ;RESET UP A HALT FOR TRAP
  1232  015752  022737  033665  000474        CMP     #<2*9$+1>,BUFF-4 ;CHECK FOR VALUE ON STACK
  1233  015760  001405                        BEQ     TST101
        015762  012737  000274  000302        MOV     #274,$FATAL     ;MOVE TO MAILBOX #  *******  274  *******
        015770  005212                        INC     (R2)            ;SET MSGTYP TO FATAL ERROR
        015772  000000                        HALT                    ;WRONG VALUE ON STACK,OR WRONG $TSTNM
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 652
```

```
1234                                    .SBTTL  TEST #101 - TEST  THAT TRACE BIT TRAPS  INHIB ON TRAP INST
                                        ;*************************************************************************
                                        ;TEST 101 - TEST  THAT TRACE BIT TRAPS  INHIB ON TRAP INST
                                        ;*************************************************************************
 015774  032737  000001  177766  TST101: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 016002  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 016004  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 016012  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 016014  000000                          HALT                    ;CPU POWER BIT FOUND SET
 016016  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 016024  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 016030  022737  000101  000304          CMP     #101,$TESTN     ;SEQUENCE ERROR?
 016036  001027                          BNE     3$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1235 016040  012706  000500              MOV     #BUFF,R6
1236 016044  012737  016102  000014      MOV     #1$,14          ;TRACE TRAP
1237 016052  005027  000016              CLR     #16
1238 016056  005027  000022              CLR     #22
1239 016062  012737  016130  000020      MOV     #4$,20          ;IOT TRAP
1240 016070  012746  000020              MOV     #20,-(SP)       ;PUSH T BIT
1241 016074  012746  016102              MOV     #1$,-(SP)       ;PUSH PC
1242 016100  000006                      RTT
1243 016102  000004              1$:     IOT                     ;TRAP, NEW CC HAVE TRACE RESET
1244 016104                      2$:
 016104  012737  000275  000302          MOV     #275,$FATAL     ;MOVE TO MAILBOX #  *******  275  *******
 016112  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 016114  000000                          HALT                    ;TRACE TRAP WAS NOT INHIBITED
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 750
1245 016116                      3$:
 016116  012737  000276  000302          MOV     #276,$FATAL     ;MOVE TO MAILBOX #  *******  276  *******
 016124  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 016126  000000                          HALT                    ;WRONG TSTNM,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 743
1246 016130  012737  000016  000014  4$:     MOV     #16,14
1247 016136  012737  000022  000020          MOV     #22,20
```

```
1248                                    .SBTTL  TEST #102 - TEST THAT THE TRACE BIT IS SAVED IN THE STACK
                                        ;******************************************************************************
                                        ;TEST 102 - TEST THAT THE TRACE BIT IS SAVED IN THE STACK
                                        ;******************************************************************************
     016144  032737  000001  177766  TST102: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     016152  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     016154  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     016162  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     016164  000000                          HALT                    ;CPU POWER BIT FOUND SET
     016166  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     016174  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     016200  022737  000102  000304          CMP     #102,$TESTN     ;SEQUENCE ERROR?
     016206  001020                          BNE     2$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1249 016210  012706  000500                  MOV     #BUFF,R6        ;SET UP STACK POINTER
1250 016214  012737  016262  000014          MOV     #3$,14          ;TRACE TRAP RETURN
1251 016222  005037  000016                  CLR     16
1252 016226  012746  000020                  MOV     #20,-(SP)       ;SET THE T BIT
1253 016232  012746  016240                  MOV     #1$,-(SP)
1254 016236  000002                          RTI
1255 016240  033727  000476  000020  1$:     BIT     BUFF-2,#20      ;CHECK FOR T BIT ON STACK
1256 016246  001005                          BNE     3$
     016250                          2$:
     016250  012737  000277  000302          MOV     #277,$FATAL     ;MOVE TO MAILBOX #  *******  277  *******
     016256  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     016260  000000                          HALT                    ;T BIT NOT SAVED ON THE STACK, OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 752
1257 016262  012737  000016  000014  3$:     MOV     #16,14
1258
1259
1260                                    ;THIS ROUTINE TEST THAT NO LEGAL ADDRESS TRAPS.
1261                                    ;AND THAT AN ILLEGAL ADDRESS TRAPS TO LOCATION 4
```

```
1262                                    .SBTTL  TEST #103 - TEST  NON-EXISTENT ADDRESS TRAPS
                                        ;******************************************************************************
                                        ;TEST 103 - TEST  NON-EXISTENT ADDRESS TRAPS
                                        ;******************************************************************************
016270  032737  000001  177766  TST103: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
016276  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
016300  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
016306  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
016310  000000                          HALT                    ;CPU POWER BIT FOUND SET
016312  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
016320  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
016324  022737  000103  000304          CMP     #103,$TESTN     ;SEQUENCE ERROR?
016332  001402                          BEQ     1000$           ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
016334  000137  016700                  JMP     18$             ;JUMP TO ERROR HALT ON SEQ ERROR
016340                          1000$:
1263
1264                                    ;THIS ROUTINE TESTS MEMORY UNTIL IT DOES A NXM TRAP
1265 016340  000402                      BR      5$
1266 016342  000000                  3$:     0
1267 016344  000000                  4$:     0
1268 016346  005000                  5$:     CLR     R0
1269 016350  005037  177766                  CLR     CPUERR          ;CLEAR CPU ERROR REGISTER
1270 016354  005037  000006                  CLR     6
1271 016360  012737  016414  000004          MOV     #8$,4           ;SET UP ADDRESS TRAP ENTRANCE
1272 016366  012706  000500          6$:     MOV     #BUFF,SP
1273 016372  105720                          TSTB    (R0)+           ;IF OUTSIDE OF CORE, TRAP TO 4
1274 016374  020027  160000                  CMP     R0,#160000      ;IS POINTER IN SIDE CORE
1275 016400  101772                          BLOS    6$              ;TEST THE REST OF CORE
1276 016402                          7$:
016402  012737  000300  000302          MOV     #300,$FATAL     ;MOVE TO MAILBOX #  *******  300  *******
016410  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
016412  000000                          HALT                    ;SHOULD HAVE TRAPPED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 751
1277                                    ;RETURN HERE ON AN ADDRESS TRAP
1278 016414  010037  016344      8$:     MOV     R0,4$           ;MOVE THE FIRST NXM LOCAVION IN 5$
1279 016420  013737  177766  000502      MOV     CPUERR,RCPUER   ;SAVE CPU ERROR REGISTER
1280 016426  042737  177413  000502      BIC     #CERMSK,RCPUER  ;MASK OFF UNUSED ERROR REG BITS
1281 016434  020027  160000              CMP     R0,#160000      ;WHICH CPU ERROR REG BIT SHOULD BE
1282                                                            ;  SET - NON EXISTANT MEMORY (BIT 5)
1283                                                            ;  OR UNIBUS TIMEOUT (BIT 4)
1284 016440  103012                      BHIS    9$              ;BRANCH IF UNIBUS TIMEOUT BIT SHOULD BE SET
1285 016442  022737  000040  000502      CMP     #40,RCPUER      ;IS NON-EXISTANT MEMORY BIT SET?
1286 016450  001417                      BEQ     10$
016452  012737  000301  000302          MOV     #301,$FATAL     ;MOVE TO MAILBOX #  *******  301  *******
016460  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
016462  000000                          HALT                    ;INCORRECT CPU ERROR REG CONTENTS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 725
1287 016464  000411                      BR      10$
1288 016466  022737  000020  000502  9$: CMP     #20,RCPUER      ;IS UNIBUS TIMEOUT BIT SET?
1289 016474  001405                      BEQ     10$
016476  012737  000302  000302          MOV     #302,$FATAL     ;MOVE TO MAILBOX #  *******  302  *******
016504  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
016506  000000                          HALT                    ;INCORRECT CPU ERROR REG CONTENTS
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 713
```

```
1290                                        ;THIS ROUTINE DOES NXM TRAPS UNTIL IT FINDS AN EXISTANT MEMORY LOCATION
1291 016510  012700  160001          10$:    MOV     #160001,R0      ;SET UP THE HIGHEST MEM LOCATION
1292 016514  005037  177766          11$:    CLR     CPUERR          ;CLEAR CPU ERROR REGISTER
1293 016520  012737  016560  000004          MOV     #14$,4          ;SET UP THE VECTOR
1294 016526  012706  000500                  MOV     #BUFF,SP
1295 016532  105740                          TSTB    -(R0)           ;DOES IT EXIST?
1296 016534  005200                  13$:    INC     R0              ;IF YES INCREMENT IT
1297 016536  020037  016344                  CMP     R0,4$           ;IS IT THE SAME LOCATION?
1298 016542  001463                          BEQ     19$
     016544  012737  000303  000302          MOV     #303,$FATAL     ;MOVE TO MAILBOX #  *******  303  *******
     016552  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     016554  000000                          HALT                    ;CONTENTS OF R0 AND 5$ SHOULD HAVE BEEN EQUAL
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 670
1299                                                                 ;IF THIS COMPARISON FAILS IT MEANS
1300                                                                 ;THAT SOME LEGAL ADDRESS TRAPPED OR
1301                                                                 ;THAT AN ILLEGAL ADDRESS DID NOT TRAP
1302 016556  000455                          BR      19$
1303
1304 016560  005737  177776          14$:    TST     STATUS
1305 016564  001405                          BEQ     15$
     016566  012737  000304  000302          MOV     #304,$FATAL     ;MOVE TO MAILBOX #  *******  304  *******
     016574  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     016576  000000                          HALT                    ;NEW PSW SHOULD HAVE BEEN ZERO
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 657
1306
1307 016600  013737  177766  000502  15$:    MOV     CPUERR,RCPUER   ;SAVE CPU ERROR REGISTER
1308 016606  042737  177413  000502          BIC     #CERMSK,RCPUER  ;MASK OFF UNUSED ERROR REG BITS
1309 016614  020027  160000                  CMP     R0,#160000      ;WHICH CPU ERROR REG BIT SHOULD BE
1310                                                                 ;  SET - NON EXISTANT MEMORY (BIT 5)
1311                                                                 ;  OR UNIBUS TIOEOUT (BIT 4)
1312 016620  103012                          BHIS    16$             ;BRANCH IF UNIBUS TIMEOUT BIT SHOULD BE SET
1313 016622  022737  000040  000502          CMP     #40,RCPUER      ;IS NON-EXISTENT MEMORY BIT SET?
1314 016630  001417                          BEQ     17$
     016632  012737  000305  000302          MOV     #305,$FATAL     ;MOVE TO MAILBOX #  *******  305  *******
     016640  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     016642  000000                          HALT                    ;INCORRECT CPU ERROR REG CONTENTS
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 635
1315 016644  000411                          BR      17$
1316 016646  022737  000020  000502  16$:    CMP     #20,RCPUER      ;IS UNIBUS TIMEOUT BIT SET?
1317 016654  001405                          BEQ     17$
     016656  012737  000306  000302          MOV     #306,$FATAL     ;MOVE TO MAILBOX #  *******  306  *******
     016664  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     016666  000000                          HALT                    ;INCORRECT CPU ERROR REG CONTENTS
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 623
1318 016670  023727  000474  016534  17$:    CMP     BUFF-4,#13$
1319 016676  001706                          BEQ     11$
     016700                          18$:
     016700  012737  000307  000302          MOV     #307,$FATAL     ;MOVE TO MAILBOX #  *******  307  *******
     016706  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     016710  000000                          HALT                    ;OLD PC WAS NOT SAVED OR WRONG $TESTN
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 612
1320 016712  012737  000006  000004  19$:    MOV     #6,4
```

```
1321 016720  005037  000006                    CLR     6
1322                                ;THIS ROUTINE WILL FIGURE OUT IF YOU HAVE A DL11W
1323
1324 016724  005037  016750                    CLR     PROFTE          ;INITIALLY CLEAR THE LOCATION
1325 016730  012706  000500                    MOV     #BUFF,SP        ;SET UP THE STACK POINTER
1326 016734  012737  016752  000C04            MOV     #DL11W,4        ;SET UP THE TRAP VECTOR
1327 016742  005737  177564                    TST     TPS             ;TEST THE PUNCH STATUS REGISTER
1328 016746  000403                            BR      DL11W1          ;BRANCH IF IT EXISTS
1329 016750  000000                PROFTE:     .WORD   0
1330 016752  005237  016750        DL11W:      INC     PROFTE          ;INCREMENT IF NO DL11W
1331 016756  012737  000006  000004  DL11W1:   MOV     #6,4
```

```
                                .SBTTL  TEST #104 - TEST THAT A TTY INRUP CAUSES AN OVERFLOW TRAP
                                ;******************************************************************************
                                ;TEST 104 - TEST THAT A TTY INRUP CAUSES AN OVERFLOW TRAP
                                ;******************************************************************************
016764  032737  000001  177766  TST104: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
016772  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
016774  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
017002  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
017004  000000                          HALT                    ;CPU POWER BIT FOUND SET
017006  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
017014  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
017020  022737  000104  000304          CMP     #104,$TESTN     ;SEQUENCE ERROR?
017026  001031                          BNE     1$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1333 017030  005737  016750            TST     PROFTE
1334 017034  001042                    BNE     3$
1335 017036  000005                    RESET
1336 017040  012737  000340  177776    MOV     #340,STATUS     ;LOCK OUT INTERRUPT
1337 017046  012706  000400            MOV     #400,R6         ;SET UP STACK TO OVERFLOW
1338 017052  012737  017124  000004    MOV     #2$,4           ;SET UP OVERFLOW TRAP
1339 017060  012737  017112  000064    MOV     #1$,64          ;SET UP INTERRUPT VECTOR
1340 017066  012737  000100  177564    MOV     #100,TTCSR      ;SET INTERRUPT ENABLE
1341 017074  005037  177776            CLR     STATUS          ;ALLOW INTERRUPT TO OCCUR
1342 017100  012737  000310  000302    MOV     #310,$FATAL     ;MOVE TO MAILBOX #  *******  310  *******
017106  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
017110  000000                          HALT                    ;NO INTERRUPT OCCURRED
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 746
1343 017112                     1$:
017112  012737  000311  000302          MOV     #311,$FATAL     ;MOVE TO MAILBOX #  *******  311  *******
017120  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
017122  000000                          HALT                    ;OVERFLOW TRAP DID NOT OCCUR OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 741
1344 017124  005037  177564     2$:    CLR     TTCSR           ;CLEAR INTERRUPT ENABLE
1345 017130  012737  000006  000004    MOV     #6,4
1346 017136  005037  000006            CLR     6
1347 0*7142                     3$:
```

```
1348                                    .SBTTL  TEST #105 - TEST THAT A TRAP OCCURS BEFORE INRUPT
                                        ;*****************************************************************************
                                        ;TEST 105 - TEST THAT A TRAP OCCURS BEFORE INRUPT
                                        ;*****************************************************************************
     017142  032737  000001  177766  TST105: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     017150  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     017152  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     017160  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     017162  000000                          HALT                    ;CPU POWER BIT FOUND SET
     017164  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     017172  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     017176  022737  000105  000304          CMP     #105,$TESTN     ;SEQUENCE ERROR?
     017204  001037                          BNE     2$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1349 017206  005737  016750                  TST     PROFTE
1350 017212  001046                          BNE     4$
1351 017214  012706  000500                  MOV     #BUFF,R6
1352 017220  012737  000340  177776          MOV     #340,STATUS     ;SET TO A HIGH PRIORITY LEVEL
1353 017226  012737  017272  000064          MOV     #1$,64
1354 017234  012737  000100  177564          MOV     #100,TTCSR      ;INTERRUPT FOR TTY PUNCH/PRINTER
1355 017242  012737  017316  000034          MOV     #3$,34          ;TRAP VECTOR
1356 017250  012737  017304  000064          MOV     #2$,64          ;TTY VECTOR
1357 017256  012737  000340  000036          MOV     #340,36         ;IF TRAP TRAPS, MOVE 340 TO PRIORITY
1358 017264  005037  177776                  CLR     STATUS          ;SHOULD INTERRUPT AT END OF CLR INST
1359 017270  104400                          TRAP                    ;TTY INTERRUPT SHOULD OVERRIDE TRAP
1360 017272                             1$:
     017272  012737  000312  000302          MOV     #312,$FATAL     ;MOVE TO MAILBOX #  *******  312  *******
     017300  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     017302  000000                          HALT                    ;NEITHER TRAP NOR INRUPT OCCURED
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 740
1361 017304                             2$:
     017304  012737  000313  000302          MOV     #313,$FATAL     ;MOVE TO MAILBOX #  *******  313  *******
     017312  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     017314  000000                          HALT                    ;INRUPT OCCURRED FIRST,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 733
1362 017316  005037  000036             3$:     CLR     36
1363 017322  042737  000100  177564          BIC     #100,TTCSR
1364 017330                             4$:
```

TEST #106 - TEST THAT A PENDING INRUP, INRUP BETWEEN TRAPS

```
                                        .SBTTL  TEST #106 - TEST THAT A PENDING INRUP, INRUP BETWEEN TRAPS
                                        ;******************************************************************************
                                        ;TEST 106 - TEST THAT A PENDING INRUP, INRUP BETWEEN TRAPS
                                        ;******************************************************************************
017330  032737  000001  177766  TST106: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
017336  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
017340  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
017346  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
017350  000000                          HALT                    ;CPU POWER BIT FOUND SET
017352  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
017360  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
017364  022737  000106  000304          CMP     #106,$TESTN     ;SEQUENCE ERROR?
017372  001031                          BNE     2$              ;BRANCH TO ERROR HALT ON SEQ ERROR
 366 017374  005737  016750             TST     PROFTE
1367 017400  001046                     BNE     4$
1368 017402  012706  000500             MOV     #BUFF,R6
1369 017406  012737  000340  177776     MOV     #340,STATUS
1370 017414  012737  000100  177564     MOV     #100,TTCSR
1371 017422  012737  017454  000034     MOV     #1$,34          ;TRAP
1372 017430  012737  017470  000064     MOV     #3$,64          ;TTY OUTPUT
1373 017436  012737  017456  000020     MOV     #2$,20          ;IOT
1374 017444  012737  000340  000022     MOV     #340,22         ;IOT PRIORITY
1375 017452  104400                     TRAP                    ;THE ACT OF TRAPPING LOWER PRIORITY
1376 017454  000004             1$:     IOT                     ;INTERRUPT SHOULD OCCUR IN PLACE OF IOT TRAP
1377 017456                     2$:
017456  012737  000314  000302          MOV     #314,$FATAL     ;MOVE TO MAILBOX #  *******  314  *******
017464  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
017466  000000                          HALT                    ;NO INTERRUPT BETWEEN TRAPS,OR WRONG $TSTNM
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 741
1378 017470  005037  000022     3$:     CLR     22              ;CLR IOT PRIORITY
1379 017474  012737  000036  000034     MOV     #36,34
1380 017502  012737  000066  000064     MOV     #66,64
1381 017510  012737  000022  000020     MOV     #22,20
1382 017516                     4$:
1383
```

```
1384                                    .SBTTL  TEST #107 - TEST THAT 'RESET' GOES TO OUTSIDE WORLD
                                        ;*************************************************************************
                                        ;TEST 107 - TEST THAT 'RESET' GOES TO OUTSIDE WORLD
                                        ;*************************************************************************
     017516  032737  000001  177766  TST107: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     017524  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     017526  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     017534  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     017536  000000                          HALT                    ;CPU POWER BIT FOUND SET
     017540  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     017546  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     017552  022737  000107  000304          CMP     #107,$TESTN     ;SEQUENCE ERROR?
     017560  001027                          BNE     TST110-12               ;BRANCH TO ERROR HALT ON SEQ ERROR
1385 017562  005737  016750                  TST     PROFTE
1386 017566  001031                          BNE     2$
1387 017570  012737  000100  177564          MOV     #100,TTCSR      ;SET INTERRUPT ENABLE
1388 017576  012737  000100  177560          MOV     #100,TRCSR      ;SET INTERRUPT ENABLE
1389 017604  000005                          RESET                   ;SHOULD CLEAR INTERRUPT ENABLE
1390 017606  032737  000100  177564          BIT     #100,TTCSR      ;TEST FOR CLEAR
1391 017614  001405                          BEQ     1$
     017616  012737  000315  000302          MOV     #315,$FATAL     ;MOVE TO MAILBOX #  *******  315  *******
     017624  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     017626  000000                          HALT                    ;RESET FAILED TO CLEAR TTCSR
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 754
1392 017630  032737  000100  177560  1$:     BIT     #100,TRCSR      ;TEST FOR CLEAR
1393 017636  001405                          BEQ     TST110
     017640  012737  000316  000302          MOV     #316,$FATAL     ;MOVE TO MAILBOX #  *******  316  *******
     017646  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     017650  000000                          HALT                    ;RESET FAILED TO CLEAR TRCSR,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 743
1394 017652                             2$:
```

```
1395                                    .SBTTL  TEST #110 - TEST THAT RESET HAS NO EFFECT ON  TRACE TRAP
                                        ;*************************************************************************
                                        ;TEST 110 - TEST THAT RESET HAS NO EFFECT ON  TRACE TRAP
                                        ;*************************************************************************
     017652  032737  000001  177766  TST110: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     017660  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     017662  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     017670  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     017672  000000                          HALT                    ;CPU POWER BIT FOUND SET
     017674  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     017702  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     017706  022737  000110  000304          CMP     #110,$TESTN     ;SEQUENCE ERROR?
     017714  001014                          BNE     2$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1396 017716  012706  000500                  MOV     #BUFF,R6        ;SET STACK
1397 017722  012737  017760  000014          MOV     #3$,14          ;SET UP TRACE VECTOR
1398 017730  012746  000020                  MOV     #20,-(R6)       ;SET THE T-BIT ON STACK
1399 017734  012746  017742                  MOV     #1$,-(R6)       ;MOVE NEW PC ON STACK
1400 017740  000006                          RTT
1401 017742  000005                  1$:     RESET                   ;SHOULD HAVE NO EFFECT
1402 017744  000005                          RESET                   ;NO EFFECT
1403 017746                          2$:
     017746  012737  000317  000302          MOV     #317,$FATAL     ;MOVE TO MAILBOX #  *******  317  *******
     017754  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     017756  000000                          HALT                    ;TRACE TRAP FAILED,OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 756
1404 017760  005037  177776          3$:     CLR     STATUS          ;CLEAR TRACK
1405 017764  005037  000016                  CLR     16              ;TRACE STATUS
1406 017770  012737  000016  000014          MOV     #16,14
1407
```

```
                                        .SBTTL  TEST #111 - TEST THAT WHEN TTY INRUPTS IT POPS NEW STATUS
                                        ;*****************************************************************************
                                        ;TEST 111 - TEST THAT WHEN TTY INRUPTS IT POPS NEW STATUS
                                        ;************************************************************ *************
 017776  032737  000001  177766  TST111: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 020004  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 020006  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 020014  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 020016  000000                          HALT                    ;CPU POWER BIT FOUND SET
 020020  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 020026  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 020032  022737  000111  000304          CMP     #111,$TESTN     ;SEQUENCE ERROR?
 020040  001051                          BNE     4$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1409 020042  005737  016750              TST     PROFTE
1410 020046  001055                      BNE     6$
1411 020050  000005                      RESET
1412 020052  012706  000500              MOV     #BUFF,R6        ;SET UP STACK
1413 020056  012737  020102  000064      MOV     #1$,64          ;INTERRUPT VECTOR
1414 020064  005037  177776              CLR     STATUS          ;DROP PROCESSOR PRIORITY
1415 020070  012737  000357  000066      MOV     #357,66         ;HIGH PRIORITY ON INTERRUPT
1416 020076  005137  177564              COM     TTCSR           ;SHOULD SET INTERRUPT ENABLE & INTERRUPT
1417 020102  023727  177776  000357  1$: CMP     STATUS,#357
1418 020110  001405                      BEQ     2$
 020112  012737  000320  000302          MOV     #320,$FATAL     ;MOVE TO MAILBOX #  *******  320  *******
 020120  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 020122  000000                          HALT                    ;INTERRUPT DID NOT POP CORRECT STATUS
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 746
1419 020124  000005              2$:     RESET                   ;CLR NTERRUPT ENABLE
1420 020126  012706  000500              MOV     #BUFF,R6        ;STACK SET UP
1421 020132  012737  020156  000064      MOV     #3$,64          ;INTERRUPT VECTOR
1422 020140  005037  000066              CLR     66              ;CLR NEW STATUS
1423 020144  012737  000157  177776      MOV     #157,STATUS     ;PROCESSOR STATUS
1424 020152  005137  177564              COM     TTCSR           ;SET INTERRUPT ENABLE
1425 020156  005737  177776      3$:     TST     STATUS
1426 020162  001405                      BEQ     5$
 020164                          4$:
 020164  012737  000321  000302          MOV     #321,$FATAL     ;MOVE TO MAILBOX #  *******  321  *******
 020172  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 020174  000000                          HALT                    ;INCORRECT STATUS,OR WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 721
1427 020176  005037  177564      5$:     CLR     TTCSR
1428 020202                      6$:
```

```
1429                                    .SBTTL  TEST #112 - TEST THE 'WAIT' INSTRUCTION
                                        ;******************************************************************************
                                        ;TEST 112 - TEST THE 'WAIT' INSTRUCTION
                                        ;******************************************************************************
      020202  032737  000001  177766  TST112: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
      020210  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
      020212  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
      020220  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      020222  000000                          HALT                    ;CPU POWER BIT FOUND SET
      020224  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
      020232  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
      020236  022737  000112  000304          CMP     #112,$TESTN     ;SEQUENCE ERROR?
      020244  001060                          BNE     6$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1430  020246  005737  016750                  TST     PROFTE
1431  020252  001064                          BNE     8$
1432  020254  042737  000100  177564          BIC     #100,TPS        ;CLEAR INTERRUPT ENABLE
1433  020262  012706  000500                  MOV     #BUFF,SP        ;SET UP THE STACK
1434  020266  012737  020356  000064          MOV     #4$,64          ;SET UP THE INTERRUPT VECTOR
1435  020274  005037  000066                  CLR     66
1436  020300  105737  177564          1$:     TSTB    TPS             ;WAIT FOR READY
1437  020304  100375                          BPL     1$              ;TO BE UP
1438  020306  012737  000015  177566          MOV     #15,TPB         ;DO A CARRIAGE RETURN
1439  020314  105737  177564          2$:     TSTB    TPS             ;WAIT FOR READY TO COME UP
1440  020320  100375                          BPL     2$
1441  020322  012737  000015  177566          MOV     #15,TPB         ;DO ANOTHER CARRIAGE RETURN
1442  020330  052737  000100  177564          BIS     #100,TPS        ;SET THE INTERRUPT ENABLE
1443  020336  005037  177776                  CLR     STATUS          ;CLEAR THE PSW
1444  020342  000001                  3$:     WAIT                    ;WAIT FOR THE INTERRUPT
1445  020344  012737  000322  000302          MOV     #322,$FATAL     ;MOVE TO MAILBOX #  *******  322  *******
      020352  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      020354  000000                          HALT                    ;WAIT INSTRUCTION DID NOT LOOP
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 733
1446  020356  005737  177776          4$:     TST     STATUS          ;IS THE PSW CORRECT?
1447  020362  001405                          BEQ     5$
      020364  012737  000323  000302          MOV     #323,$FATAL     ;MOVE TO MAILBOX #  *******  323  *******
      020372  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      020374  000000                          HALT                    ;NEW PSW SHOULD HAVE BEEN ZERO
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 723
1448  020376  023727  000474  020344  5$:     CMP     BUFF-4,#3$+2    ;IS THE OLD PC SAVED
1449  020404  001405                          BEQ     7$
      020406                          6$:
      020406  012737  000324  000302          MOV     #324,$FATAL     ;MOVE TO MAILBOX #  *******  324  *******
      020414  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      020416  000000                          HALT                    ;OLD PC WAS NOT SAVED OR WRONG $TESTN
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 712
1450  020420  005037  177564          7$:     CLR     TPS             ;CLEAR INTERRUPT ENABLE
1451  020424                          8$:
```

```
                                        .SBTTL  TEST #113 - TEST THAT ALL RESERVED INS TRAP
                                        ;*************************************************************************
                                        ;TEST 113 - TEST THAT ALL RESERVED INS TRAP
                                        ;*************************************************************************
 020424  032737  000001  177766  TST113: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 020432  001410                         BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 020434  012737  000177  000302         MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 020442  005212                         INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 020444  000000                         HALT                    ;CPU POWER BIT FOUND SET
 020446  042737  000001  177766         BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 020454  005237  000304         100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 020460  022737  000113  000304         CMP     #113,$TESTN     ;SEQUENCE ERROR?
 020466  001154                         BNE     RET4            ;BRANCH TO ERROR HALT ON SEQ ERROR
1453 020470  012737  020516  000244         MOV     #TRP244,244     ; SET UP TO SEE IF
1454 020476  013737  000010  020544         MOV     10,TENSAV       ;  THIS PROCESSOR HAS THE
1455 020504  012737  020526  000010         MOV     #TRAP10,10      ;  FLOATING POINT OPTION
1456 020512  170007                         .WORD   170007          ; AN ILLEGAL FPP INSTRUCTION
1457 020514  000415                         BR      TSFCIS
1458 020516  013737  021150  021154  TRP244: MOV     FPP,FINISH      ;IF FPP IN-- RESET END OF TABLE POINTER
1459 020524  000002                         RTI                     ; AND RETURN
1460 020526                         TRAP10:                         ; LEAVE THE TABLE ALONE
1461 020526  005737  000306                 TST     $PASS           ;FIRST PASS??
1462 020532  001003                         BNE     1$              ;BRANCH IF NO
1463 020534  004737  024216                 JSR     PC,PRTMSG       ;PRINT MESSAGE POINTED TO BY NEXT WORD
1464 020540  024036                         .WORD   MSGNFP          ;USING THIS ADDRESS AS THE START OF THE MESSAGE
1465 020542  000002                 1$:     RTI                     ; RETURN
1466 020544  000000                 TENSAV: .WORD   0               ; A PLACE TO STORE CONTENTS OF 10
1467 020546  000000                 SAVSTK: .WORD   0               ;LOCATION TO SAVE STACK POINTER
1468 020550                         TSFCIS:                 ;SEE IF PROCESSOR HAS CIS OPTION
1469 020550  012737  020636  000004         MOV     #AROUND,4       ;SET TIME OUT TRAP VECTOR
1470
1471 020556  012737  020610  000010         MOV     #TNCIS,10       ;SET UP RESERVE INST TRAP VECTOR
1472 020564  012700  160000                 MOV     #160000,R0      ;POINT R0 TO NON-EXISTED MEMORY LOC.
1473 020570  010637  020546                 MOV     SP,SAVSTK       ;MOVE STACK POINTER VALUE TO SAVSTK AND
1474 020574  162737  000004  020546         SUB     #4,SAVSTK       ;CORRECT TO SHOW 2 EXPECTED PUSHES
1475 020602  076020                         .WORD   76020           ;CIS INST = L2DR (DESTROYS CONTENTS OF R0,R1,R2,R3)
1476 020604  000000                         HALT                    ;CIS INST FAILED TO TRAP
1477 020606  000430                         BR      ADJNC
1478 020610                         TNCIS:  ;NO CIS OPTION,EXPECTED TRAP ETHER  TO 4 OR 10 DID NOT HAPPEN
1479
1480 020610  005737  000306                 TST     $PASS           ;FIRST PASS
1481 020614  001003                         BNE     1$
1482 020616  004737  024216                 JSR     PC,PRTMSG       ;PRINT MESSAGE POINTED TO BY THE NEXT WORD
1483 020622  024101                         .WORD   MSGNCIS         ;USE THIS ADDRESS TO PRINT THE MESSAGE
1484 020624  012703  021044         1$:     MOV     #TABLE1,TAB
1485 020630  062716  000002                 ADD     #2,(SP)         ;CORRECT POINTER TO AFTER THE HALT
1486 020634  000002                         RTI                     ;RETURN
1487 020636  012703  021074         AROUND: MOV     #TABLE,TAB      ;CIS OPTION PRESENT
1488 020642  020637  020546                 CMP     SP,SAVSTK       ;SEE IF 2 WORDS WERE PUSHED ON THE STACK
1489 020646  001405                         BEQ     1$              ;BRANCH IF SO - CIS SWITCH IN NORMAL POSITION
1490 020650  004737  024216                 JSR     PC,PRTMSG       ;GO PRINT THE MESSAGE POINTED TO BY THE NEXT WORD
1491 020654  023600                         .WORD   CISMSG          ;USE THIS ADDRESS TO PRINT THE MESSAGE
1492 020656  000000                         HALT                    ;HALT, ALLOWING USER TO PUT SWITCH IN NORMAL POSITION
1493 020660  000733                         BR      TSFCIS          ;GO BACK TO CHECK AGAIN
1494 020662  062716  000002         1$:     ADD     #2,(SP)         ;CORRECT POINTER TO AFTER THE HALT
1495 020666  000002                         RTI                     ;RETURN
1496 020670  012737  000246  000244 ADJNC:  MOV     #246,244        ;  RESTORE THE TRAP VECTOR
```

```
1497 020676  013737  020544  000010          MOV     TENSAV,10       ; RESTORE THE ILLEGAL INST. VECTOR
1498 020704  012305                  GIN1:   MOV     (TAB)+,FIRST    ;FIRST OR CURRENT INSTRUCTION
1499 020706  012301                          MOV     (TAB)+,LAST     ;LAST INSTRUCTION OR GROUP
1500 020710  020537  021154                  CMP     FIRST,FINISH    ;TESTED ALL
1501 020714  001525                          BEQ     GIN3            ;YES BRANCH
1502 020716  010537  021156                  MOV     FIRST,INST      ;SET UP INST
1503 020722  005237  021156          GIN2:   INC     INST
1504 020726  012737  020750  000010          MOV     #RET,10         ;SET UP RETURN FROM TRAP
1505 020734  012706  000500                  MOV     #BUFF,SP        ;SET UP STACK POINTER
1506 020740  005037  177776                  CLR     CC              ;CLEAR PRIORITY
1507 020744  000137  021156                  JMP     INST            ;EXECUTE RESERVED INSTRUCTION
1508
1509                                 ;TRAPPING SHOULD SEND YOU HERE
1510 020750  020627  000474          RET:    CMP     SP,#BUFF-4      ;TEST DECREMENT OF SP
1511 020754  001405                          BEQ     RET1
1512 020756  012737  000325  000302          MOV     #325,$FATAL     ;MOVE TO MAILBOX #  *******  325  *******
     020764  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     020766  000000                          HALT                    ;WRONG DECREMENT
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 637
1513 020770  023727  000474  021160  RET1:   CMP     BUFF-4,#INST+2  ;LOC OF INST UNINCREMENTED
1514 020776  001405                          BEQ     RET2
1515 021000  012737  000326  000302          MOV     #326,$FATAL     ;MOVE TO MAILBOX #  *******  326  *******
     021006  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021010  000000                          HALT                    ;INST INC ON TRAP
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 626
1516 021012  005737  000476          RET2:   TST     BUFF-2
1517 021016  001405                          BEQ     RET3
     021020                          RET4:
     021020  012737  000327  000302          MOV     #327,$FATAL     ;MOVE TO MAILBOX #  *******  327  *******
     021026  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021030  000000                          HALT                    ;CONDITION CODES SET ON TRAP OR WRONG $TSTNM
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 616
1518 021032  023701  021156          RET3:   CMP     INST,LAST
1519 021036  001722                          BEQ     GIN1            ;SET UP NEW GROUP
1520 021040  000137  020722                  JMP     GIN2            ;FINISH OLD GROUP
1521
1522 021044  076017                  TABLE1: 76017                   ;CIS INSTRUCTIONS
1523 021046  076032                          76032
1524 021050  076037                          76037
1525 021052  076045                          76045
1526 021054  076047                          76047
1527 021056  076077                          76077
1528 021060  076117                          76117
1529 021062  076132                          76132
1530 021064  076137                          76137
1531 021066  076145                          76145
1532 021070  076147                          76147
1533 021072  076177                          76177
1534 021074  000007                  TABLE:  7
1535 021076  000077                          77
1536 021100  000207                          207                     ;RTS,RT1,JMP
1537 021102  000227                          227
1538 021104  007077                          7077
1539 021106  007777                          7777
```

```
1540 021110  075037                          075037
1541 021112  076017                          76017
1542 021114  076032                          76032
1543 021116  076037                          76037
1544 021120  076045                          76045
1545 021122  076047                          76047
1546 021124  076132                          76132
1547 021126  076137                          76137
1548 021130  076145                          76145
1549 021132  076147                          76147
1550 021134  076077                          76077
1551 021136  076117                          76117
1552 021140  106377                          106377
1553 021142  106477                          106477
1554 021144  106677                          106677
1555 021146  107777                          107777
1556 021150  167777                FPP:      167777          ;  START OF THE FPP INSTRUCTIONS
1557 021152  177777                          177777
1558 021154  021154                FINISH:   .               ;END FLAG
1559 021156  000000                INST:     HALT            ;WILL CONTINUE RESERVED INST
1560 021160  000000                          HALT            ;SHOULD TRAP TO LOC 10
1561 021162  000000                          HALT            ;LOC 10 SHOULD SEND YOU TO
1562 021164  000000                          HALT            ;RET
1563 021166  000000                          HALT
1564
1565 021170                        GIN3:
```

```
1566                                    .SBTTL  TEST #114 - TEST ILLEGAL HALT
                                        ;******************************************************************************
                                        ;TEST 114 - TEST ILLEGAL HALT
                                        ;******************************************************************************
     021170  032737  000001  177766  TST114: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     021176  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     021200  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     021206  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021210  000000                          HALT                    ;CPU POWER BIT FOUND SET
     021212  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     021220  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     021224  022737  000114  000304          CMP     #114,$TESTN     ;SEQUENCE ERROR?
     021232  001073                          BNE     4$              ;BRANCH TO ERROR HALT ON SEQ ERROR
1567 021234  012706  000500                  MOV     #BUFF,SP        ;STACK POINTER SETUP
1568 021240  005037  177766                  CLR     CPUERR          ;CLEAR CPU ERROR REGISTER
1569 021244  012737  021300  000004          MOV     #1$,RTRAP5      ;SETUP TRAP RETURN
1570 021252  052737  040000  177776          BIS     #040000,PSW     ;GO TO SUPER MODE
1571 021260  000000                          HALT                    ;EXECUTE INST UNDER TEST
1572                                    ;FAILURE, NO TRAP
1573 021262  105037  177777                  CLRB    PSW+1           ;GO BACK TO KERNEL
1574 021266  012737  000330  000302          MOV     #330,$FATAL     ;MOVE TO MAILBOX #  *******  330  *******
     021274  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021276  000000                          HALT                    ;HALT IN SUPER MODE FAILED TO TRAP
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 755
1575 021300                             1$:
1576 021300  013737  177766  000502          MOV     CPUERR,RCPUER   ;READ AND SAVE CPU ERROR REGISTER
1577 021306  042737  177413  000502          BIC     #CERMSK,RCPUER  ;MASK OFF UNUSED CPU ERR REG BITS
1578 021314  022737  000200  000502          CMP     #200,RCPUER     ;IS ILLEGAL HALT BIT SET?
1579 021322  001405                          BEQ     2$
     021324  012737  000331  000302          MOV     #331,$FATAL     ;MOVE TO MAILBOX #  *******  331  *******
     021332  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021334  000000                          HALT                    ;INCORRECT CPU ERR REG CONTENTS
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 736
1580 021336  005037  177766          2$:     CLR     CPUERR          ;CLEAR CPU ERR REG
1581 021342  012737  021376  000004          MOV     #3$,RTRAP5      ;SETUP TRAP RETURN
1582 021350  052737  140000  177776          BIS     #140000,PSW     ;GO TO USER MODE
1583 021356  000000                          HALT
1584                                    ;FAILURE, NO TRAP
1585 021360  105037  177777                  CLRB    PSW+1           ;GO BACK TO KERNEL
1586 021364  012737  000332  000302          MOV     #332,$FATAL     ;MOVE TO MAILBOX #  *******  332  *******
     021372  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021374  000000                          HALT                    ;HALT IN USER MODE FAILED TO TRAP
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 716
1587 021376                             3$:
1588 021376  013737  177766  000502          MOV     CPUERR,RCPUER   ;SAVE CPU ERROR REGISTER
1589 021404  042737  177413  000502          BIC     #CERMSK,RCPUER  ;MASK OFF UNUSED CPU ERR REG BITS
1590 021412  022737  000200  000502          CMP     #200,RCPUER     ;IS ILLEGAL HALT BIT SET?
1591 021420  001405                          BEQ     5$
     021422                             4$:
     021422  012737  000333  000302          MOV     #333,$FATAL     ;MOVE TO MAILBOX #  *******  333  *******
     021430  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021432  000000                          HALT                    ;INCORRECT CPU ERR REG CONTENTS
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 677
```

```
1592 021434                        5$:
1593 021434  005037  177766              CLR     CPUERR
1594 021440  105037  177777              CLRB    PSW+1          ;GO BACK TO KERNEL MODE
```

```
1595                                    .SBTTL  TEST #115 - TEST SPL INST. FOR NOP IN USER/SUPER MODES
                                        ;******************************************************************************
                                        ;TEST 115 - TEST SPL INST. FOR NOP IN USER/SUPER MODES
                                        ;******************************************************************************
     021444  032737  000001  177766  TST115: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
     021452  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
     021454  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
     021462  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021464  000000                          HALT                    ;CPU POWER BIT FOUND SET
     021466  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
     021474  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
     021500  022737  000115  000304          CMP     #115,$TESTN     ;SEQUENCE ERROR?
     021506  001125                          BNE     SEQ             ;BRANCH TO ERROR HALT ON SEQ ERROR
1596 021510  012706  000500                  MOV     #BUFF,SP        ;SETUP STACK
1597 021514  052737  040000  177776          BIS     #040000,PSW     ;GO TO SUPER MODE
1598 021522  000277                          SCC                     ;SET CC
1599 021524  000231                          SPL     1               ;SPL SHOULD=NOP IN USER/SUPER MODES
1600 021526  000232                          SPL     2
1601 021530  000233                          SPL     3
1602 021532  000234                          SPL     4
1603 021534  000235                          SPL     5
1604 021536  000236                          SPL     6
1605 021540  000237                          SPL     7
1606 021542  013737  177776  022456          MOV     PSW,SPSW        ;SAVE PSW
1607 021550  023727  022456  040017          CMP     SPSW,#040017    ;VARIFY THAT PSW HAS NOT CHANGED
1608 021556  001407                          BEQ     1$
1609 021560  105037  177777                  CLRB    PSW+1           ;GO BACK TO KERNEL
1610 021564  012737  000334  000302          MOV     #334,$FATAL     ;MOVE TO MAILBOX #  *******  334  *******
     021572  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021574  000000                          HALT                    ;PRIORTY LEVELS CHANGE
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 744
1611 021576  012737  040340  177776  1$:     MOV     #040340,PSW     ;SET PRIORTY TO 7
1612 021604  000257                          CCC                     ;CLEAR CONDITION CODES
1613 021606  000230                          SPL     0               ;SPL SHOULD=NOP IN SUPERVISOR MODE
1614 021610  013737  177776  022456          MOV     PSW,SPSW        ;SAVE PSW
1615 021616  023727  022456  040340          CMP     SPSW,#040340    ;VARIFY THAT PSW PRIORTY AND CONDITION CODES HAVE NOT CHANGE
1616 021624  001407                          BEQ     2$
1617 021626  105037  177777                  CLRB    PSW+1           ;GO BACK TO KERNEL
1618 021632  012737  000335  000302          MOV     #335,$FATAL     ;MOVE TO MAILBOX #  *******  335  *******
     021640  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021642  000000                          HALT                    ;SPL INSTRUCTION CHANDED PSW SHOULD BE NOP
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 721
1619 021644  012737  140000  177776  2$:     MOV     #140000,PSW     ;GO TO USER MODE
1620 021652  000277                          SCC                     ;SET CC
1621 021654  000231                          SPL     1               ;SPL SHOULD=NOP IN USER MODE
1622 021656  000232                          SPL     2
1623 021660  000233                          SPL     3
1624 021662  000234                          SPL     4
1625 021664  000235                          SPL     5
1626 021666  000236                          SPL     6
1627 021670  000237                          SPL     7
1628 021672  013737  177776  022456          MOV     PSW,SPSW        ;SAVE PSW
1629 021700  023727  022456  140017          CMP     SPSW,#140017    ;VARIFY THAT PSW HAS NOT CHANGED
1630 021706  001407                          BEQ     3$
1631 021710  105037  177777                  CLRB    PSW+1           ;GO BACK TO KERNEL
```

```
1632 021714  012737  000336  000302          MOV     #336,$FATAL     ;MOVE TO MAILBOX #  *******  336  *******
     021722  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021724  000000                          HALT                    ;PRIORTY LEVELS HAS CHANGED
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 670
1633 021726  012737  140340  177776  3$:     MOV     #140340,PSW     ;SET PRIORTY TO 7
1634 021734  000257                          CCC                     ;CLEAR CONDITION CODES
1635 021736  000230                          SPL     0               ;SPL SHOULD=NOP IN USER MODE
1636 021740  013737  177776  022456          MOV     PSW,SPSW        ;SAVE PSW
1637 021746  023727  022456  140340          CMP     SPSW,#140340    ;VARIFY THAT PSW PRIORTY AND CONDITION CODES HAVE NOT CHANGE
1638 021754  001407                          BEQ     FSPL
1639 021756  105037  177777                  CLRB    PSW+1           ;GO BACK TO KERNEL
1640 021762                          SEQ:
     021762  012737  000337  000302          MOV     #337,$FATAL     ;MOVE TO MAILBOX #  *******  337  *******
     021770  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     021772  000000                          HALT                    ;SPL INST.CHANGED PSW OR WRONG TEST#
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 645
1641 021774                          FSPL:
1642 021774  105037  177777                  CLRB    PSW+1           ;GO BACK TO KERNEL
1643                                         ;THIS TEST VARIFIES FOR ALL COMBINATIONS OF PIR AND PROCESSOR
1644                                         ;PRIORTY LEVELS THAT REQUESTS ARE GRANTED (TRAP TO 240 OCCURS
1645                                         ;BY THE PROCESSOR,ONLY WHEN THE PIR IS AT A HIGHER LEVEL THAN
1646                                         ;THE PROCESSOR.
1647                                         ;THE CONTENTS OF SPIR,SPSW AND TRP240 SHOULD BE EXAMINED ON ERROR.
1648                                         ;SPIR BITS 2-0 CONTAINS ONE LESS THAN THE PIR REQUEST LEVEL AT
1649                                         ;THE TIME OF ERROR.SPSW BITS 2-0 CONTAINS THE PROCESSOR PRIORTY
1650                                         ;AT THE TIME OF ERROR.TRP240 INDICATES WHETHER OR NOT A TRAP WAS
1651                                         ;EXPECTED (1= EXPECTING TRAP TO 240)
1652                                         ;THE SPL INSTRUCTION IS USED TO SETUP PROCESSOR PRIORTY.
1653                                         ;NOTE: THIS IS THE FIRST REAL TEST OF THE SPL INST.
1654                                         ;ON ERROR,IF EXPECTED PIRQ TRAP DID NOT OCCURE VARIFY SPL
1655                                         ;OPERATION BY COMPARING SPSW BITS<2-0> WITH PSW PRIORTY
1656                                         ;BITS<7-5>.ON ERROR IF AN UNSPECTED PIRQ TRAP OCCURED
1657                                         ;VARIFY SPL OPERATION BY COMPARING SPSW BITS<2-0> WITH
1658                                         ;PROCESSOR PSW<7-5> ON STACK.
1659
```

```
1660                                    .SBTTL  TEST #116 -  TEST PIRQ LEVELS AND SPL INSTRUCTION
                                        ;******************************************************************************
                                        ;TEST 116 -  TEST PIRQ LEVELS AND SPL INSTRUCTION
                                        ;******************************************************************************
022000  032737  000001  177766  TST116: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
022006  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
022010  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
022016  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
022020  000000                          HALT                    ;CPU POWER BIT FOUND SET
022022  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
022030  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
022034  022737  000116  000304          CMP     #116,$TESTN     ;SEQUENCE ERROR?
022042  001155                          BNE     PTRP            ;BRANCH TO ERROR HALT ON SEQ ERROR
022044  005037  022460                  CLR     PIRPSW
1661 022050  012737  022412  000000    MOV     #ZTRP,0         ;SET LOCATION ZERO TRAP VEC
1662 022056  012737  000340  000002    MOV     #340,2
1663 022064  012737  022426  000004    MOV     #T4TRP,4        ;SET UP FAILURE TRAP VEC
1664 022072  012737  000340  000006    MOV     #340,6
1665 022100  012737  022442  000024    MOV     #T24TRP,24      ;SETUP POWER FAIL VEC
1666 022106  012737  000340  000026    MOV     #340,26
1667 022114  012737  022354  000240    MOV     #PQTRP,PIRVC1   ;SETUP PIRQ VEC
1668 022122  012737  000340  000242    MOV     #340,PIRVC2     ;SET 242 TO PRIORTY 7
1669 022130  012706  000500          MLOOP: MOV     #BUFF,SP        ;SETUP STACK
1675 022134  052737  000340  177776          BIS     #340,PSW        ;SET PROCESSOR PRIORTY TO 7
1676                                    ;COMPUTE EXPECTED RESULT
1677 022142  013700  022460                  MOV     PIRPSW,R0
1678 022146  042700  177707                  BIC     #177707,R0      ;SETUP SPIR
1679 022152  006200                          ASR     R0
1680 022154  006200                          ASR     R0
1681 022156  006200                          ASR     R0
1682 022160  010037  022454                  MOV     R0,SPIR         ;SAVE R0 IN SPIR
1683 022164  013700  022460                  MOV     PIRPSW,R0
1684 022170  042700  177770                  BIC     #177770,R0      ;SETUP SPSW
1685 022174  010037  022456                  MOV     R0,SPSW
1686 022200  023737  022454  022456          CMP     SPIR,SPSW
1687 022206  002003                          BGE     2$              ;BRANCH IF PIR > PSW
1688 022210  005037  022462                  CLR     TRP240          ;CLEAR FLAG
1689 022214  000403                          BR      3$
1690 022216  012737  177777  022462  2$:     MOV     #177777,TRP240  ;SET FLAG
1691 022224  004737  022464          3$:     JSR     PC,SETPIRQ      ;SETUP PIRQ BASE ON THE # IN SPIR
1692                                    ;SET PSW PRIORTY BASE ON THE # IN SPSW
1693 022230  042737  000007  022244          BIC     #7,1$           ;CLEAR LSB 3 BITS OF SPL INST.AT 1$
1694 022236  053737  022456  022244          BIS     SPSW,1$         ;SET LSB 3 BITS OF SPL INST.TO DESIDED PROC. PRI.
1695 022244  000230                  1$:     SPL     0       ;THE ACTUAL PRIORTY SET INTO THE PSW IS CONTROLLED BY
1696                                                          ;THE PREVIOUS TWO INSTRUCTION.
1697 022246  000240                          NOP
1698 022250  005737  022462                  TST     TRP240
1699 022254  001403                          BEQ     NXTST
1700 022256  012737  000340  000302          MOV     #340,$FATAL     ;MOVE TO MAILBOX #  *******  340  *******
022264  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
022266  000000                          HALT                    ;EXPECTED PIRQ TRAP BUT DID NOT GET IT
                                                                ;TO SCOPE REPLACE HALT WITH 240
                                                                ;AND REPLACE NEXT INST WITH 717
1701 022270  000240                  NXTST:  NOP                     ;SET UP FOR PASS COUNT
1702
1703
1704 022272  005237  022460                  INC     PIRPSW          ;SETUP FOR NEXT PASS THROUGH LOOP
```

TEST #116 - TEST PIRQ LEVELS AND SPL INSTRUCTION

```
1705 022276  023727  022460  000070          CMP     PIRPSW,#70
1706 022304  002711                          BLT     MLOOP           ;BRANCH TO MLOOP IF COUNT IS LESS THAN 70
1707 022306  005037  177772                  CLR     PIRQ            ;DONE WITH LOOPING PIRQ REQUESTS
1708 022312  012737  000006  000004          MOV     #6,4            ;RESTORE RETURNS
1709 022320  005037  000006                  CLR     6
1710 022324  005037  000000                  CLR     0
1711 022330  012737  024132  000024          MOV     #PWRDWN,24
1712 022336  012737  000242  000240          MOV     #242,240
1713 022344  005037  000242                  CLR     242
1714 022350  000137  022656                  JMP     TST117
1715 022354  005737  022462          PQTRP:  TST     TRP240          ;PRIORTY TRAP SERVICE
1716 022360  001343                          BNE     NXTST
     022362  012737  000341  000302          MOV     #341,$FATAL     ;MOVE TO MAILBOX #  *******  341  *******
     022370  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     022372  000000                          HALT                    ;EXPECTED NO PIRQ TRAP,BUT GOT ONE
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 655

1717 022374  000735                          BR      NXTST
1721 022376                          PTRP:
     022376  012737  000342  000302          MOV     #342,$FATAL     ;MOVE TO MAILBOX #  *******  342  *******
     022404  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     022406  000000                          HALT                    ; WRONG TEST NUMBER
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 615

1725 022410  000522                          BR      TST117
1726 022412                          ZTRP:
1727 022412  012737  000343  000302          MOV     #343,$FATAL     ;MOVE TO MAILBOX #  *******  343  *******
     022420  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     022422  000000                          HALT                    ;UNSPECTED TRAP TO ZERO OCCURED DURING PIRQ TST
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 641

1728 022424  000721                          BR      NXTST
1729 022426                          T4TRP:
1730 022426  012737  000344  000302          MOV     #344,$FATAL     ;MOVE TO MAILBOX #  *******  344  *******
     022434  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     022436  000000                          HALT                    ;UNSPECTED TRAP TO 4 DURING PIRQ TESTING
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 633

1731 022440  000713                          BR      NXTST
1732 022442                          T24TRP:
1733 022442  012737  000345  000302          MOV     #345,$FATAL     ;MOVE TO MAILBOX #  *******  345  *******
     022450  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     022452  000000                          HALT                    ;UNSPECTED POWER FAIL TRAP DURING PIRQ TESTING
                                                                     ;TO SCOPE REPLACE HALT WITH 240
                                                                     ;AND REPLACE NEXT INST WITH 625

1734 022454  000000                  SPIR:   .WORD   0
1735 022456  000000                  SPSW:   .WORD   0
1736 022460  000000                  PIRPSW: .WORD   0
```

```
                177772                          PIRQ=177772
                000240                          PIRVC1=240
                000242                          PIRVC2=242
022462          000000                          TRP240: .WORD   0


022464                                  SETPIRQ:
022464  023727  022454  000000                  CMP     SPIR,#0
022472  001435                                  BEQ     1$
022474  023727  022454  000001                  CMP     SPIR,#1
022502  001435                                  BEQ     2$
022504  023727  022454  000002                  CMP     SPIR,#2
022512  001435                                  BEQ     3$
022514  023727  022454  000003                  CMP     SPIR,#3
022522  001435                                  BEQ     4$
022524  023727  022454  000004                  CMP     SPIR,#4
022532  001435                                  BEQ     5$
022534  023727  022454  000005                  CMP     SPIR,#5
022542  001435                                  BEQ     6$
022544  023727  022454  000006                  CMP     SPIR,#6
022552  001435                                  BEQ     7$
022554  012737  000346  000302                  MOV     #346,$FATAL     ;MOVE TO MAILBOX #  *******  346  *******
022562  005212                                  INC     (R2)            ;SET MSGTYP TO FATAL ERROR
022564  000000                                  HALT                    ;# IN SPIR DOES NOT MAKE SENSE OR SPIR NOT=0-6
                                                                        ;TO SCOPE REPLACE HALT WITH 240
                                                                        ;AND REPLACE NEXT INST WITH 560
022566  012737  001000  177772  1$:             MOV     #1000,PIRQ
022574  000427                                  BR      10$
022576  012737  002000  177772  2$:             MOV     #2000,PIRQ
022604  000423                                  BR      10$
022606  012737  004000  177772  3$:             MOV     #4000,PIRQ
022614  000417                                  BR      10$
022616  012737  010000  177772  4$:             MOV     #10000,PIRQ
022624  000413                                  BR      10$
022626  012737  020000  177772  5$:             MOV     #20000,PIRQ
022634  000407                                  BR      10$
022636  012737  040000  177772  6$:             MOV     #40000,PIRQ
022644  000403                                  BR      10$
022646  012737  100000  177772  7$:             MOV     #100000,PIRQ
022654  000207                  10$:            RTS     PC


                                        ;TABLE FOR PIRQ SETUP
                                        ;SPIR   PIR LEVEL       SETPIRQ         # LOADED INTO PIRQ REG R0
                                        ;0000       1           BIT9            1000
                                        ;0001       2           BIT10           2000
                                        ;0002       3           BIT11           4000
                                        ;0003       4           BIT12           10000
                                        ;0004       5           BIT13           20000
                                        ;0005       6           BIT14           40000
                                        ;0006       7           BIT15           100000
```

```
                                        .SBTTL  TEST #117 - CHK ODD ADRS TRAP WITH SP AT ODD ADRS ON RTI
                                        ;*************************************************************************
                                        ;TEST 117 - CHK ODD ADRS TRAP WITH SP AT ODD ADRS ON RTI
                                        ;*************************************************************************
 022656  032737  000001  177766  TST117: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 022664  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
 022666  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
 022674  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 022676  000000                          HALT                    ;CPU POWER BIT FOUND SET
 022700  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
 022706  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
 022712  022737  000117  000304          CMP     #117,$TESTN     ;SEQUENCE ERROR?
 022720  001026                          BNE     OATSE           ;BRANCH TO ERROR HALT ON SEQ ERROR
1786 022722  012737  140000  177776          MOV     #140000,177776  ;SET CURRENT MODE TO USER
1787 022730  012706  000001                  MOV     #1,SP           ;PUT ODD ADDRESS IN STACK POINTER
1788 022734  012737  022752  000004          MOV     #1$,4           ;TRAPS TO 1$
1789 022742  012737  000340  000006          MOV     #340,6          ;PRIORITY TO 7
1790 022750  000002                          RTI                     ;TEST RTI - SHOULD ODD ADDRESS TRAP TO 4
1791 022752  022737  030340  177776  1$:     CMP     #30340,177776   ;SEE IF PREVIOUS MODE USER AND PRIORITY 7 LOADED
1792 022760  001413                          BEQ     ODADEN          ;BRANCH OUT IF OK
1793                                    ;*************************************************************************
1794                                    ;IF PSW CONTAINS 30000, YOU MIGHT BE MISSING M7095 ECO #6.
1795                                    ;*************************************************************************
1796 022762  012737  000347  000302          MOV     #347,$FATAL     ;MOVE TO MAILBOX #  *******  347  *******
 022770  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 022772  000000                          HALT                    ;PSW NOT PROPERLY LOADED
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 752
1797 022774  000405                          BR      ODADEN          ;BRANCH TO EXIT TEST
1798 022776                          OATSE:
 022776  012737  000350  000302          MOV     #350,$FATAL     ;MOVE TO MAILBOX #  *******  350  *******
 023004  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
 023006  000000                          HALT                    ;WRONG $TSTNM
                                                                 ;TO SCOPE REPLACE HALT WITH 240
                                                                 ;AND REPLACE NEXT INST WITH 744
1799 023010  012737  000340  177776  ODADEN: MOV     #340,177776     ;MODE BACK TO KERNEL, PRIORITY = 7
1800 023016  012737  000006  000004          MOV     #6,4            ;RESET TRAPS TO 4 VECTOR                ;DPM002
1801 023024  005037  000006                  CLR     6               ;PUT A HALT IN 6                        ;DPM0C2
```

```
 1802                                   .SBTTL  TEST #120 - PIRQ/UNIBUS INTERRUPT ORDER TEST
                                        ;******************************************************************************
                                        ;TEST 120 - PIRQ/UNIBUS INTERRUPT ORDER TEST
                                        ;******************************************************************************
      023030  032737  000001  177766  TST120: BIT     #1,CPUERR       ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
      023036  001410                          BEQ     100$            ;BRANCH AROUND CLEAR AND HALT IF CLEAR
      023040  012737  000177  000302          MOV     #177,$FATAL     ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
      023046  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      023050  000000                          HALT                    ;CPU POWER BIT FOUND SET
      023052  042737  000001  177766          BIC     #1,CPUERR       ;CLEAR THE BIT FOUND SET
      023060  005237  000304          100$:   INC     $TESTN          ;UPDATE TEST NUMBER
      023064  022737  000120  000304          CMP     #120,$TESTN     ;SEQUENCE ERROR?
      023072  001137                          BNE     PUIO            ;BRANCH TO ERROR HALT ON SEQ ERROR
 1803 023074  012706  000500                  MOV     #500,SP         ;RESET THE STACK POINTER                    ;DPM002
 1804 023100  012737  000340  177776          MOV     #340,177776     ;PRIORITY TO 7                              ;DPM002
 1805 023106  013737  000100  023460          MOV     100,CVECSV      ;SAVE THE CLOCK VECTOR                      ;DPM002
 1806 023114  012737  023404  000100          MOV     #PUIOEN,100     ;SET CLOCK INTERRUPTS TO THE END            ;DPM002
 1807 023122  013737  000102  023462          MOV     102,CPRISV      ;SAVE THE CLOCK PRIORITY                    ;DPM002
 1808 023130  012737  000300  000102          MOV     #300,102        ;MOVE PRIORIY 6 TO CLOCK PRIORITY           ;DPM002
 1809 023136  013737  000240  023464          MOV     240,PVECSV      ;SAVE THE PIRQ VECTOR                       ;DPM002
 1810 023144  012737  023240  000240          MOV     #3$,240         ;SET PIRQ INTERRUPTS TO 3$                  ;DPM002
 1811 023152  013737  000242  023466          MOV     242,PPRISV      ;SAVE THE PIRQ PRIORITY                     ;DPM002
 1812 023160  012737  000040  000242          MOV     #40,242         ;SET PIRQ PRIORITY TO 1                     ;DPM002
 1813 023166  012737  001000  177772          MOV     #1000,PIRQ      ;REQUEST PIRQ LEVEL 1 INTERRUPT             ;DPM002
 1814 023174  105737  177546          1$:     TSTB    177546          ;IS THE LINE CLOCK READY YET                ;DPM002
 1815 023200  100375                          BPL     1$              ;BRANCH BACK UNTIL IT IS                    ;DPM002
 1816 023202  012737  000100  177546          MOV     #100,177546     ;SET THE INTERRUPT ENABLE, CLEAR READY      ;DPM002
 1817 023210  105737  177546          2$:     TSTB    177546          ;WAIT FOR READY AGAIN                       ;DPM002
 1818 023214  100375                          BPL     2$              ;BRANCH BACK UNTIL SET AGAIN                ;DPM002
 1819 023216  005037  177776                  CLR     177776          ;LOWER PRIORITY TO 0                        ;DPM002
 1820 023222  000240                          NOP                     ;WAIT OUT THE 11/44 DELAY                   ;D-M002
 1821 023224  012737  000351  000302          MOV     #351,$FATAL     ;MOVE TO MAILBOX #  *******  351  *******
      023232  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      023234  000000                          HALT                    ;NO INTERRUPT WHEN EXPECTED
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 716
 1822 023236  000405                          BR      NXTPRT          ;BRANCH TO NEXT PART                        ;DPM002
 1823 023240                          3$:
 1824 023240  012737  000352  000302          MOV     #352,$FATAL     ;MOVE TO MAILBOX #  *******  352  *******
      023246  005212                          INC     (R2)            ;SET MSGTYP TO FATAL ERROR
      023250  000000                          HALT                    ;INTERRUPT WRONGLY GIVEN TO THE PIRQ
                                                                      ;TO SCOPE REPLACE HALT WITH 240
                                                                      ;AND REPLACE NEXT INST WITH 710
 1825                                   ;******************************************************************************
 1826                                   ; YOU MAY BE MISSING M7098 ECO # 7
 1827                                   ;******************************************************************************
 1828 023252  012737  000340  177776  NXTPRT: MOV     #340,177776     ;SET PRIORITY BACK TO 7                     ;DPM002
 1829 023260  012706  000500                  MOV     #500,SP         ;RESET THE STACK POINTER                    ;DPM002
 1830 023264  012737  000300  000242          MOV     #300,242        ;SET PIRQ PRIORITY TO 6                     ;DPM002
 1831 023272  012737  023360  000100          MOV     #3$,100         ;CLOCK VECTOR TO ERROR ROUTINE              ;DPM002
 1832 023300  012737  023404  000240          MOV     #PUIOEN,240     ;PIRQ INTERRUPTS TO PUIOEN                  ;DPM002
 1833 023306  012737  001000  177772          MOV     #1000,PIRQ      ;REQUEST PIRQ LEVEL 1 INTERRUPT             ;DPM002
 1834 023314  105737  177546          1$:     TSTB    177546          ;IS THE LINE CLOCK READY YET                ;DPM002
 1835 023320  100375                          BPL     1$              ;BRANCH BACK UNTIL IT IS                    ;DPM002
 1836 023322  012737  000100  177546          MOV     #100,177546     ;SET THE INTERRUPT ENABLE, CLEAR READY      ;DPM002
 1837 023330  105737  177546          2$:     TSTB    177546          ;WAIT FOR READY AGAIN                       ;DPM002
 1838 023334  100375                          BPL     2$              ;BRANCH BACK UNTIL SET AGAIN                ;DPM002
```

```
1839 023336  005037  177776           CLR     177776          ;LOWER PRIORITY TO 0                    ;DPM002
1840 023342  000240                   NOP                     ;WAIT OUT THE 11/44 DELAY               ;DPM002
1841 023344  012737  000353  000302   MOV     #353,$FATAL     ;MOVE TO MAILBOX #  *******  353  *******
     023352  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     023354  000000                   HALT                    ;NO INTERRUPT WHEN EXPECTED
                                                              ;TO SCOPE REPLACE HALT WITH 240
                                                              ;AND REPLACE NEXT INST WITH 646
1842 023356  000412                   BR      PUIOEN          ;BRANCH OVER SEQUENCE ERROR CODE        ;DPM002
1843 023360                   3$:
     023360  012737  000354  000302   MOV     #354,$FATAL     ;MOVE TO MAILBOX #  *******  354  *******
     023366  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     023370  000000                   HALT                    ;INTERRUPT WRONGLY GIVEN TO THE CLOCK
                                                              ;TO SCOPE REPLACE HALT WITH 240
                                                              ;AND REPLACE NEXT INST WITH 640
1844 023372                   PUIO:
     023372  012737  000355  000302   MOV     #355,$FATAL     ;MOVE TO MAILBOX #  *******  355  *******
     023400  005212                   INC     (R2)            ;SET MSGTYP TO FATAL ERROR
     023402  000000                   HALT                    ;WRONG $TSTNM
                                                              ;TO SCOPE REPLACE HALT WITH 240
                                                              ;AND REPLACE NEXT INST WITH 633
1845 023404  005037  177546   PUIOEN: CLR     177546          ;CLEAR THE CLOCK CSR                    ;DPM002
1846 023410  012737  000340  177776   MOV     #340,177776     ;SET PRIORITY BACK TO 7                 ;DPM002
1847 023416  005037  177772           CLR     PIRQ            ;CLEAR THE PIRQ REGISTER                ;DPM002
1848 023422  013737  023460  000100   MOV     CVECSV,100      ;RESTORE THE CLOCK VECTOR               ;DPM002
1849 023430  013737  023462  000102   MOV     CPRISV,102      ;RESTORE THE CLOCK PRIORITY             ;DPM002
1850 023436  013737  023464  000240   MOV     PVECSV,240      ;RESTORE THE PIRQ VECTOR                ;DPM002
1851 023444  013737  023466  000242   MOV     PPRISV,242      ;RESTORE THE PIRQ PRIORITY              ;DPM002
1852 023452  012706  000500           MOV     #500,SP         ;CLEAN UP STACK                         ;DPM002
1853 023456  000404                   BR      END             ;BRANCH OVER THE STORAGE LOCATIONS      ;DPM002
1854
1855 023460  000000           CVECSV: .WORD   0               ;LOCATION TO STORE CLOCK VECTOR         ;DPM002
1856 023462  000000           CPRISV: .WORD   0               ;LOCATION TO STORE CLOCK PRIORITY       ;DPM002
1857 023464  000000           PVECSV: .WORD   0               ;LOCATION TO STORE PIRQ VECTOR          ;DPM002
1858 023466  000000           PPRISV: .WORD   0               ;LOCATION TO STORE PIRQ PRIORITY        ;DPM002
```

END OF PASS ROUTINE

```
1859                                            .SBTTL  END OF PASS ROUTINE
1860 023470  005237  000306          END:       INC     $PASS
1861 023474  105237  023576                     INCB    PASSPT          ;SHOULD PRINT THIS PASS?
1862 023500  001017                             BNE     ACT             ;NO
1863 023502  132737  000040  000321             BITB    #40,$ENVM       ;WILL APT ALLOW PRINTING?
1864 023510  001013                             BNE     ACT             ;NO
1865 023512  023727  000042  023550             CMP     42,#$ENDAD
1866 023520  001407                             BEQ     ACT
1867 023522  004737  024216                     JSR     PC,PRTMSG       ;PRINT MESSAGE POINTED TO BY THE NEXT WORD
1868 023526  023723                             .WORD   MSG             ;USE THIS ADDRESS TO PRINT THE MESSAGE
1869 023530  000005                             RESET
1870 023532  012737  177761  023576             MOV     #-15.,PASSPT    ;DO IT ABOUT 15 DECIMAL TIMES
1871 023540  013700  000042          ACT:       MOV     42,R0           ;CHECK ACT
1872 023544  001405                             BEQ     GOAGIN          ;KEEP GOING IF NO ACT
1873 023546  000005                             RESET
1874 023550  004710                  $ENDAD:    JSR     PC,(R0)         ;ACT HOOKS
1875 023552  000240                             NOP
1876 023554  000240                             NOP
1877 023556  000240                             NOP
1878 023560  012737  000012  000010  GOAGIN:    MOV     #12,10
1879 023566  005037  000012                     CLR     12
1880 023572  000137  000540                     JMP     RESTRT          ;DO NEXT PASS
1881 023576  177777                  PASSPT:    -1
```

ASCII MESSAGES

```
1882                                            .SBTTL  ASCII MESSAGES
1883 023600  015  012  103  CISMSG: .ASCII  <15><12>.CIS SWITCH IS IN THE MAINTENANCE POSITION.<15><12>
1884 023655  120  125  124          .ASCIZ  .PUT IN NORMAL POSITION AND CONTINUE.<15><12>
1885 023723  015  012  105  MSG:    .ASCIZ  <15><12>.END OF CKKABDO 11/44 TRAPS.
1886 023760  015  012  103  PNAME:  .ASCIZ <15><12>+CKKABDO 11/44 TRAPS+
1887 024006  015  012  103  TITLE:  .ASCIZ <15><12>+CKKABDO 11/44 TRAPS+<15><12>
1888 024036  015  012  116  MSGNFP: .ASCIZ  <15><12>.NO FLOATING POINT OPTION PRESENT.
1889 024101  015  012  116  MSGNCIS:.ASCIZ  <15><12>.NO CIS OPTION PRESENT .
1890                                            .EVEN
```

POWER DOWN AND POWER UP ROUTINES

```
                                                .SBTTL  POWER DOWN AND POWER UP ROUTINES
1892 024132  012737  024142  000024  PWRDWN:   MOV     #PWRUP,24
1893 024140  000000                           HALT
1894
1895 024142  012737  024132  000024  PWRUP:    MOV     #PWRDWN,24
1896 024150  012706  000>00                    MOV     #BUFF,SP
1897 024154  132737  000040  000321           BITB    #40,$ENVM       ;WILL APT ALLOW PRINTING?
1898 024162  001003                           BNE     PFRES           ;NO
1899 024164  004737  024216                   JSR     PC,PRTMSG       ;PRINT MESSAGE POINTED TO BY THE NEXT WORD
1900 024170  024176                           .WORD   MSGPWF          ;USE THIS ADDRESS TO PRINT THE MESSAGE
1901 024172  000137  000540          PFRES:   JMP     RESTRT
1902 024176     015     012     120  MSGPWF:  .ASCIZ  <15><12>.POWER FAILED!.
```

```
1903                                            .SBTTL  SUBROUTINE TO PRINT A MESSAGE
1904 024216  017600  000000          PRTMSG:  MOV     @0(SP),R0       ;MOVE MESSAGE START ADDRESS TO R0
1905 024222  062716  000002                   ADD     #2,(SP)         ;FUDGE RETURN OVER STARTING ADDRESS PARAMETER
1906 024226  132737  000040  000321           BITB    #40,$ENVM       ;WILL APT ALLOW PRINTING?
1907 024234  001011                           BNE     1$              ;BRANCH IF NO
1908 024236  105737  177564          2$:      TSTB    TPS             ;TTY READY
1909 024242  100375                           BPL     2$              ;NO WAIT
1910 024244  112037  177566                   MOVB    (R0)+,TPB       ;PRINT CHARACTER
1911 024250  001372                           BNE     2$              ;NEXT IF NOT DONE.
1912 024252  105737  177564          3$:      TSTB    TPS
1913 024256  100375                           BPL     3$
1914 024260  000207                  1$:      RTS     PC
```

```
   1915          000001                          .END
```

SYMBOL TABLE

```
ABASE = 000000
ACDW1 = 000000
ACDW2 = 000000
ACPUOP= 000000
ACT     023540
ADDW0 = 000000
ADDW1 = 000000
ADDW10= 000000
ADDW11= 000000
ADDW12= 000000
ADDW13= 000000
ADDW14= 000000
ADDW15= 000000
ADDW2 = 000000
ADDW3 = 000000
ADDW4 = 000000
ADDW5 = 000000
ADDW6 = 000000
ADDW7 = 000000
ADDW8 = 000000
ADDW9 = 000000
ADEVCT= 000000
ADEVM = 000000
ADJNC   020670
AENV  = 000000
AENVM = 000000
AFATAL= 000000
AMADR1= 000000
AMADR2= 000000
AMADR3= 000000
AMADR4= 000000
AMAMS1= 00000C
AMAMS2= 000000
AMAMS3= 000000
AMAMS4= 000000
AMSGAD= 000000
AMSGLG= 000000
AMSGTY= 000000
AMTYP1= 000000
AMTYP2= 000000
AMTYP3= 000000
AMTYP4= 000000
APASS = 000000
APRIOR= 000000
AROUND  020636
ASWREG= 000000
ATESTN= 000000
AUNIT = 000000
AUSWR = 000000
AVECT1= 000000
AVECT2= 000000
BEGIN   000510
BELL  = 000240
BUFF    000500
CC    - 177776
CERMSK= 177413
CISMSG  023600
CPRISV  023462
CPUERR= 177766
CVECSV  023460
DL11W   016752
DL11W1  016756
END     023470
FINISH  021154
FIRST =%000005
FPP     021150
FSPL    021774
GIN1    020704
GIN2    020722
GIN3    021170
GOAGIN  023560
HERE  = 000000
HLT   = 000000
ILLA  = 004700
ILLB  = 000100
INST    021156
KTSTA   000506
KTVEC   000504
K1      000626
K10     000644
K11     000646
K12     000650
K2      000630
K3      000632
K4      000634
K5      000636
K6      000640
K7      000642
LAST  =%000001
MLOOP   022130
MSG     023723
MSGNCI  024101
MSGNFP  024036
MSGPWF  024176
NOP   = 000240
NXTPRT  023252
NXTST   022270
OATSE   022776
ODADEN  023010
PASSPT  023576
PFRES   024172
PIRPSW  022460
PIRQ  = 177772
PIRVC1= 000240
PIRVC2= 000242
PNAME   023760
PPRISV  023466
PQTRP   022354
PROFTE  016750
PRTMSG  024216
PSW   = 177776
PTRP    022376
PUIO    023372
PUIOEN  023404
PVECSV  023464
PWRDWN  024132
PWRUP   024142
RCPUER  000502
RESTRT  000540
RET     020750
RET1    020770
RET2    021012
RET3    021032
RET4    021020
RTRAP = 000010
RTRAP1= 000034
RTRAP2= 000020
RTRAP3= 000030
RTRAP4= 000014
RTRAP5= 000004
R6    =%000006
R7    =%000007
SAVSTK  020546
SEQ     021762
SETPIR  022464
SPIR    022454
SPSW    022456
STATUS= 177776
TAB   =%000003
TABLE   021074
TABLE1  021044
TENSAV  020544
TITLE   024006
TNCIS   020610
TPB   = 177566
TPS   = 177564
TRAPA = 000010
TRAP10  020526
TRCSR = 177560
TRP240  022462
TRP244  020516
TRT   = 000003
TSFCIS  020550
TST1    000652
TST10   002620
TST100  015452
TST101  015774
TST102  016144
TST103  016270
TST104  016764
TST105  017142
TST106  017330
TST107  017516
TST11   003012
TST110  017652
TST111  017776
TST112  020202
TST113  020424
TST114  021170
TST115  021444
TST116  022000
TST117  022656
TST12   003350
TST120  023030
TST13   003454
TST14   003554
TST15   003656
TST16   004032
TST17   004356
TST2    001232
TST20   004504
TST21   004576
TST22   004676
TST23   005000
TST24   005156
TST25   005514
TST26   005606
TST27   005706
TST3    001612
TST30   006010
TST31   006166
TST32   006506
TST33   006634
TST34   006726
TST35   007026
TST36   007130
TST37   007306
TST4    002102
TST40   007644
TST41   007736
TST42   010036
TST43   010140
TST44   010316
TST45   010636
TST46   010730
TST47   011030
TST5    002324
TST50   011132
TST51   011310
TST52   011626
TST53   012010
TST54   012112
TST55   012216
TST56   012414
TST57   012740
TST6    002416
TST60   013100
TST61   013200
TST62   013364
TST63   013472
TST64   013600
TST65   013706
TST66   014014
TST67   014122
TST7    002516
TST70   014250
TST71   014356
TST72   014514
TST73   014620
TST74   014744
TST75   015056
TST76   015166
TST77   015326
TTCSR = 177564
T24TRP  022442
T4TRP   022426
ZTRP    022412
$APTHD  000330
$CPUOP  000326
$DEVCT  000310
$ENDAD  023550
$ENV    000320
$ENVM   000321
$ERN  = 000356
$ERROR= 000302
$ETABL  000320
$ETEND  000330
$FATAL  000302
$HIBTS  000330
$MAIL   000300
$MBADR  000332
$MSGAD  000314
$MSGLG  000316
$MSGTY  000300
$PASS   000306
$PASTM  000336
$SVPC = 000300
$SWR  = 000000
$SWREG  000322
$TESTN  000304
$TN   = 000121
$TSTM   000334
$TSTNM= 000304
$UNIT   000312
$UNITM  000340
$USWR   000324
$X    = 023074
$XX   = 177634
$XXX  = 000633
$Y    = 022044
$YY   = 022130
.$X   = 000330
```

. ABS. 024262 000

SYMBOL TABLE

000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 8483 WORDS (34 PAGES)
DYNAMIC MEMORY: 9474 WORDS (36 PAGES)
ELAPSED TIME: 00:17:03
KKABD.BIN,CKKABD/CR/-SP/NL:TOC=CKKABD.MLB/ML,CKKABD.P11

SYMBOL	VALUE	REFERENCES			
ABASE	= 000000	17-393			
ACDW1	= 000000	17-393			
ACDW2	= 000000	17-393			
ACPUOP	= 000000	17-393	17-393		
ACT	023540	103-1862	103-1864	103-1866	#103-1871
ADDW0	= 000000	17-393			
ADDW1	= 000000	17-393			
ADDW10	= 000000	17-393			
ADDW11	= 000000	17-393			
ADDW12	= 000000	17-393			
ADDW13	= 000000	17-393			
ADDW14	= 000000	17-393			
ADDW15	= 000000	17-393			
ADDW2	= 000000	17-393			
ADDW3	= 000000	17-393			
ADDW4	= 000000	17-393			
ADDW5	= 000000	17-393			
ADDW6	= 000000	17-393			
ADDW7	= 000000	17-393			
ADDW8	= 000000	17-393			
ADDW9	= 000000	17-393			
ADEVCT	= 000000	17-393	17-393		
ADEVM	= 000000	17-393			
ADJNC	020670	96-1477	#96-1496		
AENV	= 000000	17-393	17-393		
AENVM	= 000000	17-393	17-393		
AFATAL	= 000000	17-393	17-393		
AMADR1	= 000000	17-393			
AMADR2	= 000000	17-393			
AMADR3	= 000000	17-393			
AMADR4	= 000000	17-393			
AMAMS1	= 000000	17-393			
AMAMS2	= 000000	17-393			
AMAMS3	= 000000	17-393			
AMAMS4	= 000000	17-393			
AMSGAD	= 000000	17-393	17-393		
AMSGLG	= 000000	17-393	17-393		
AMSGTY	= 000000	17-393	17-393		
AMTYP1	= 000000	17-393			
AMTYP2	= 000000	17-393			
AMTYP3	= 000000	17-393			
AMTYP4	= 000000	17-393			
APASS	= 000000	17-393	17-393		
APRIOR	= 000000	17-393			
AROUND	020636	96-1469	#96-1487		
ASWREG	= 000000	17-393	17-393		
ATESTN	= 000000	17-393	17-393		
AUNIT	= 000000	17-393	17-393		
AUSWR	= 000000	17-393	17-393		
AVECT1	= 000000	17-393			
AVECT2	= 000000	17-393			
BEGIN	000510	16-386	16-389	#20-405	

SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES								
BELL	= 000240	#14-363								
BUFF	000500	#19-401	26-564	27-570	27-573	28-576	28-579	29-582	29-587	29-589
		29-594	30-599	30-610	31-627	32-633	32-636	33-639	33-642	34-645
		34-650	34-652	34-656	35-659	35-670	36-686	37-695	38-701	38-704
		39-707	39-710	40-713	40-718	40-720	40-725	41-728	41-739	42-754
		43-760	43-763	44-766	44-769	45-772	45-777	45-779	45-784	46-787
		46-798	47-812	48-822	49-828	49-831	50-834	50-837	51-840	51-845
		51-848	51-853	52-856	52-867	53-889	54-895	54-898	55-901	55-904
		56-907	56-912	56-914	56-919	57-922	57-933	58-945	59-951	59-954
		60-957	60-960	61-964	61-969	61-971	61-976	62-979	62-990	63-1006
		64-1018	64-1021	65-1024	65-1027	66-1032	66-1037	66-1039	66-1044	67-1047
		67-1058	79-1141	80-1150	80-1157	81-1160	81-1165	82-1169	83-1180	84-1196
		85-1208	85-1212	85-1214	85-1218	85-1220	85-1224	85-1226	85-1232	86-1235
		87-1249	87-1255	88-1272	88-1294	88-1318	88-1325	90-1351	91-1368	93-1396
		94-1412	94-1420	95-1433	95-1448	96-1505	96-1510	96-1513	96-1516	97-1567
		98-1596	99-1675	105-1896						
CC	= 177776	*14-370	*29-584	*29-591	30-608	30-618	*34-647	*34-654	35-668	35-679
		*40-715	*40-722	41-737	41-747	*45-774	*45-781	46-796	46-807	*51-842
		*51-850	52-865	52-875	*56-909	*56-916	57-931	57-941	*61-966	*61-973
		62-988	62-998	*66-1034	*66-1041	67-1056	67-1066	*96-1506		
CERMSK	= 177413	#14-372	63-1012	68-1077	88-1280	88-1308	97-1577	97-1589		
CISMSG	023600	96-1491	#104-1883							
CPRISV	023462	*102-1807	102-1849	#102-1856						
CPUERR	= 177766	#14-371	22-436	*22-436	23-486	*23-486	24-526	*24-526	25-549	*25-549
		26-563	*26-563	27-569	*27-569	28-575	*28-575	29-581	*29-581	30-598
		*30-598	31-624	*31-624	32-632	*32-632	33-638	*33-638	34-644	*34-644
		35-658	*35-658	36-683	*36-683	37-694	*37-694	38-700	*38-700	39-706
		*39-706	40-712	*40-712	41-727	*41-727	42-753	*42-753	43-759	*43-759
		44-765	*44-765	45-771	*45-771	46-786	*46-786	47-809	*47-809	48-821
		*48-821	49-827	*49-827	50-833	*50-833	51-839	*51-839	52-855	*52-855
		53-888	*53-868	54-894	*54-894	55-900	*55-900	56-906	*56-906	57-921
		*57-921	58-944	*58-944	59-950	*59-950	60-956	*60-956	61-963	*61-963
		62-978	*62-978	63-1002	*63-1002	*63-1003	63-1004	63-1011	*63-1015	64-1017
		*64-1017	65-1023	*65-1023	66-1029	*66-1029	67-1046	*67-1046	68-1069	*68-1069
		*68-1070	68-1076	*68-1080	69-1081	*69-1081	70-1087	*70-1087	71-1114	*71-1114
		72-1115	*72-1115	73-1116	*73-1116	74-1117	*74-1117	75-1118	*75-1118	76-1119
		*76-1119	77-1122	*77-1122	78-1124	*78-1124	79-1140	*79-1140	80-1149	*80-1149
		81-1159	*81-1159	82-1168	*82-1168	83-1178	*83-1178	84-1195	*84-1195	85-1207
		*85-1207	86-1234	*86-1234	87-1248	*87-1248	88-1262	*88-1262	*88-1269	88-1279
		*88-1292	88-1307	89-1332	*89-1332	90-1348	*90-1348	91-1365	*91-1365	92-1384
		*92-1384	93-1395	*93-1395	94-1408	*94-1408	95-1429	*95-1429	96-1452	*96-1452
		97-1566	*97-1566	*97-1568	97-1576	*97-1580	97-1588	*97-1593	98-1595	*98-1595
		99-1660	*99-1660	101-1785	*101-1785	102-1802	*102-1802			
CVECSV	023460	*102-1805	102-1848	#102-1855						
DL11W	016752	88-1326	#88-1330							
DL11W1	016756	88-1328	#88-1331							
END	023470	102-1853	#103-1860							
FINISH	021154	*96-1458	96-1500	#96-1558						
FIRST	=%000005	#14-347	*96-1498	96-1500	96-1502					
FPP	021150	96-1458	#96-1556							
FSPL	021774	98-1638	#98-1641							
GIN1	020704	#96-1498	96-1519							

SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES								
GIN2	020722	#96-1503	96-1520							
GIN3	021170	96-1501	#96-1565							
GOAGIN	023560	103-1872	#103-1878							
HERE	= 000000	#21-424	*22-438	*22-443	22-483					
HLT	= 000000	#14-352								
ILLA	= 004700	#14-368	76-1119							
ILLB	= 000100	#14-369	77-1122							
INST	021156	*96-1502	*96-1503	96-1507	96-1513	96-1518	#96-1559			
KTSTA	000506	#19-404								
KTVEC	000504	#19-403								
K1	000626	#21-426	*23-488	23-489	23-492	*23-496	23-497	23-500	*23-503	23-505
		*23-511	23-513	*23-519	23-521	23-524				
K10	000644	#21-433	24-533	24-536	24-538	24-541	24-541	24-544	24-544	24-547
K11	000646	#21-434								
K12	000650	#21-435								
K2	000630	#21-427								
K3	000632	#21-428								
K4	000634	#21-429								
K5	000636	#21-430	*23-487	23-490	*23-495	*23-504	23-506	23-508	*23-512	23-514
		23-516	*23-520	23-522						
K6	000640	#21-431	23-498	24-536						
K7	000642	#21-432	24-527	24-527	24-530	24-530	24-533	24-538	24-547	
LAST	=%000001	#14-341	*96-1499	96-1518						
MLOOP	022130	#99-1675	99-1706							
MSG	023723	103-1868	#104-1885							
MSGNCI	024101	96-1483	#104-1889							
MSGNFP	024036	96-1464	#104-1888							
MSGPWF	024176	105-1900	#105-1902							
NOP	= 000240	#14-364								
NXTPRT	023252	102-1822	#102-1828							
NXTST	022270	99-1700	#99-1701	99-1716	99-1717	99-1728	99-1731			
OATSE	022776	101-1785	#101-1798							
ODADEN	023010	101-1792	101-1797	#101-1799						
PASSPT	023576	*20-406	*103-1861	*103-1870	#103-1881					
PFRES	024172	105-1898	#105-1901							
PIRPSW	022460	*99-1661	99-1678	99-1684	*99-1704	99-1705	#99-1736			
PIRQ	= 177772	*99-1707	#100-1738	*100-1760	*100-1762	*100-1764	*100-1766	*100-1768	*100-1770	*100-1772
		*102-1813	*102-1833	*102-1847						
PIRVC1	= 000240	*99-1668	#100-1739							
PIRVC2	= 000242	*99-1669	#100-1740							
PNAME	023760	#104-1886								
PPRISV	023466	*102-1811	102-1851	#102-1858						
PQTRP	022354	99-1668	#99-1715							
PROFTE	016750	*88-1324	#88-1329	*88-1330	89-1333	90-1349	91-1366	92-1385	94-1409	95-1430
PRTMSG	024216	20-409	96-1463	96-1482	96-1490	103-1867	105-1899	#106-1904		
PSW	= 177776	#14-373	*97-1570	*97-1573	*97-1582	*97-1585	*97-1594	*98-1597	98-1606	*98-1609
		*98-1611	98-1614	*98-1617	*98-1619	98-1628	*98-1631	*98-1633	98-1636	*98-1639
		*98-1642	*99-1676							
PTRP	022376	99-1660	#99-1721							
PUIO	023372	102-1802	#102-1844							
PUIOEN	023404	102-1806	102-1832	102-1842	#102-1845					
PVECSV	023464	*102-1809	102-1850	#102-1857						

SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES								
PWRDWN	024132	20-413	99-1711	#105-1892	105-1895					
PWRUP	024142	105-1892	#105-1895							
RCPUER	000502	#19-402	*63-1011	*63-1012	63-1013	*68-1076	*68-1077	68-1078	*88-1279	*88-1280
		88-1285	88-1288	*88-1307	*88-1308	88-1313	88-1316	*97-1576	*97-1577	97-1578
		*97-1588	*97-1589	97-1590						
RESTRT	000540	20-408	#20-411	103-1880	105-1901					
RET	020750	96-1504	#96-1510							
RET1	020770	96-1511	#96-1513							
RET2	021012	96-1514	#96-1516							
RET3	021032	96-1517	#96-1518							
RET4	021020	96-1452	#96-1517							
RTRAP	= 000010	#14-367	*26-565	*27-571	*28-577	*29-583	*29-590	*30-600	*30-601	*30-611
		*30-612	*53-890	*54-896	*55-902	*56-908	*56-915	*57-923	*57-924	*57-934
		*57-935	*58-946	*59-952	*60-958	*61-965	*61-972	*62-980	*62-981	*62-991
		*62-992								
RTRAP1	= 000034	#14-358	*31-628	*32-634	*33-640	*34-646	*34-653	*35-660	*35-661	*35-671
		*35-672								
RTRAP2	= 000020	#14-357	*37-696	*38-702	*39-708	*40-714	*40-721	*41-729	*41-730	*41-740
		*41-741								
RTRAP3	= 000030	#14-356	*42-755	*43-761	*44-767	*45-773	*45-780	*46-788	*46-789	*46-799
		*46-800								
RTRAP4	= 000014	#14-355	*48-823	*49-829	*50-835	*51-841	*51-849	*52-857	*52-858	*52-868
		*52-869	*79-1142	*80-1151	*81-1161					
RTRAP5	= 000004	#14-354	*63-1007	*64-1019	*65-1025	*66-1033	*66-1040	*67-1048	*67-1049	*67-1059
		*67-1060	*97-1569	*97-1581						
R6	=%000006	#14-348	*20-412	*22-437	*22-438	22-439	*22-442	*22-443	22-444	*22-447
		*22-448	*22-448	22-449	*22-452	*22-454	22-455	*22-458	*22-460	22-461
		*22-464	*22-466	*22-470	*22-472	22-473	*22-476	*22-478	*22-482	*22-483
		22-484	*23-490	*23-491	*23-498	*23-499	*23-506	*23-507	*23-514	*23-515
		*23-522	*23-523	*68-1071	*68-1073	*69-1082	*69-1084	69-1085	*70-1088	*70-1095
		*70-1097	*70-1099	*70-1101	*71-1114	*72-1115	*73-1116	*74-1117	*75-1118	*76-1119
		76-1120	*77-1122	*78-1127	*78-1128	*78-1129	*78-1130	*78-1131	*78-1132	*78-1133
		*78-1134	*78-1135	*78-1136	*85-1208	*85-1214	*85-1220	*85-1226	*86-1235	*87-1249
		*89-1337	*90-1351	*91-1368	*93-1396	*93-1398	*93-1399	*94-1412	*94-1420	
R7	=%000007	#14-350	*85-1210	*85-1216	*85-1222	*85-1229				
SAVSTK	020546	#96-1467	*96-1473	*96-1474	96-1488					
SEQ	021762	98-1595	#98-1640							
SETPIR	022464	99-1692	#100-1744							
SPIR	022454	*99-1683	99-1687	#99-1734	100-1745	100-1747	100-1749	100-1751	100-1753	100-1755
		100-1757								
SPSW	022456	*98-1606	98-1607	*98-1614	98-1615	*98-1628	98-1629	*98-1636	98-1637	*99-1686
		99-1687	99-1695	#99-1735						
STATUS	= 177776	#14-365	*25-551	*25-557	88-1304	*89-1336	*89-1341	*90-1352	*90-1358	*91-1369
		*93-1404	*94-1414	94-1417	*94-1423	94-1425	*95-1443	95-1446		
TAB	=%000003	#14-344	*96-1484	*96-1487	*96-1498	*96-1499				
TABLE	021074	96-1487	#96-1534							
TABLE1	021044	96-1484	#96-1522							
TENSAV	020544	*96-1454	#96-1466	96-1497						
TITLE	024006	20-410	#104-1887							
TNCIS	020610	96-1471	#96-1478							
TPB	= 177566	#14-362	*95-1438	*95-1441	*106-1910					
TPS	= 177564	#14-361	88-1327	*95-1432	95-1436	95-1439	*95-1442	*95-1450	106-1908	106-1912

SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES								
TRAPA	= 000010	#14-366	26-566	27-572	28-578	29-586	29-593	30-602	30-613	
TRAP10	020526	96-1455	#96-1460							
TRCSR	= 177560	#14-360	*92-1388	92-1392						
TRP240	022462	*99-1689	*99-1691	99-1699	99-1715	#100-1741				
TRP244	020516	96-1453	#96-1458							
TRT	= 000003	#14-353	48-824	49-830	50-836	51-844	51-852	52-859	52-870	75-1118
TSFCIS	020550	96-1457	#96-1468	96-1493						
TST1	000652	20-422	#22-436							
TST10	002620	28-575	28-580	#29-581						
TST100	015452	84-1195	84-1206	#85-1207						
TST101	015774	85-1207	85-1233	#86-1234						
TST102	016144	#87-1248								
TST103	016270	#88-1262								
TST104	016764	#89-1332								
TST105	017142	#90-1348								
TST106	017330	#91-1365								
TST107	017516	#92-1384								
TST11	003012	29-581	29-597	#30-598						
TST110	017652	92-1384	92-1393	#93-1395						
TST111	017776	#94-1408								
TST112	020202	#95-1429								
TST113	020424	#96-1452								
TST114	021170	#97-1566								
TST115	021444	#98-1595								
TST116	022000	#99-1660								
TST117	022656	99-1714	99-1725	#101-1785						
TST12	003350	#31-624								
TST120	023030	#102-1802								
TST13	003454	31-624	#32-632							
TST14	003554	32-632	32-637	#33-638						
TST15	003656	33-638	33-643	#34-644						
TST16	004032	34-644	34-657	#35-658						
TST17	004356	35-658	35-682	#36-683						
TST2	001232	22-436	22-485	#23-486						
TST20	004504	#37-694								
TST21	004576	37-694	#38-700							
TST22	004676	38-700	38-705	#39-706						
TST23	005000	39-706	39-711	#40-712						
TST24	005156	40-712	40-726	#41-727						
TST25	005514	#42-753								
TST26	005606	42-753	#43-759							
TST27	005706	43-759	43-764	#44-765						
TST3	001612	23-486	23-525	#24-526						
TST30	006010	44-765	44-770	#45-771						
TST31	006166	45-771	45-785	#46-786						
TST32	006506	46-786	46-808	#47-809						
TST33	006634	#48-821								
TST34	006726	48-821	#49-827							
TST35	007026	49-827	49-832	#50-833						
TST36	007130	50-833	50-838	#51-839						
TST37	007306	51-839	51-854	#52-855						
TST4	002102	24-526	24-548	#25-549						

SYMBOL CROSS REFERENCE

SYMBOL		VALUE	REFERENCES							
TST40		007644	#53-888							
TST41		007736	53-888	#54-894						
TST42		010036	54-894	54-899	#55-900					
TST43		010140	55-900	55-905	#56-906					
TST44		010316	56-906	56-920	#57-921					
TST45		010636	57-921	57-943	#58-944					
TST46		010730	58-944	#59-950						
TST47		011030	59-950	59-955	#60-956					
TST5		002324	25-549	25-562	#26-563					
TST50		011132	60-956	60-961	#61-963					
TST51		011310	61-963	61-977	#62-978					
TST52		011626	62-978	62-1000	#63-1002					
TST53		012010	#64-1017							
TST54		012112	64-1017	64-1022	#65-1023					
TST55		012216	65-1023	65-1028	#66-1029					
TST56		012414	66-1029	66-1045	#67-1046					
TST57		012740	67-1046	67-1068	#68-1069					
TST6		002416	#27-569							
TST60		013100	#69-1081							
TST61		013200	69-1081	69-1086	#70-1087					
TST62		013364	70-1087	#71-1114						
TST63		013472	#72-1115							
TST64		013600	#73-1116							
TST65		013706	#74-1117							
TST66		014014	#75-1118							
TST67		014122	#76-1119							
TST7		002516	27-569	27-574	#28-575					
TST70		014250	76-1121	#77-1122						
TST71		014356	#78-1124							
TST72		014514	#79-1140							
TST73		014620	79-1140	#80-1149						
TST74		014744	80-1149	80-1158	#81-1159					
TST75		015056	81-1159	81-1166	#82-1168					
TST76		015166	82-1168	82-1176	#83-1178					
TST77		015326	83-1178	#84-1195						
TTCSR	=	177564	#14-359	*89-1340	*89-1344	*90-1354	*90-1363	*91-1370	*92-1387	92-1390	*94-1416
			*94-1424	*94-1427						
T24TRP		022442	99-1666	#99-1732						
T4TRP		022426	99-1664	#99-1729						
ZTRP		022412	99-1662	#99-1726						
$APTHD		000330	18-395	#18-395						
$CPUOP		000326	#17-393							
$DEVCT		000310	#17-393							
$ENDAD		023550	16-391	20-407	103-1865	#103-1874				
$ENV		000320	#17-393							
$ENVM		000321	#17-393	103-1863	105-1897	106-1906				