PDP11/04

CPU TEST

MD-11-DGKAA-A

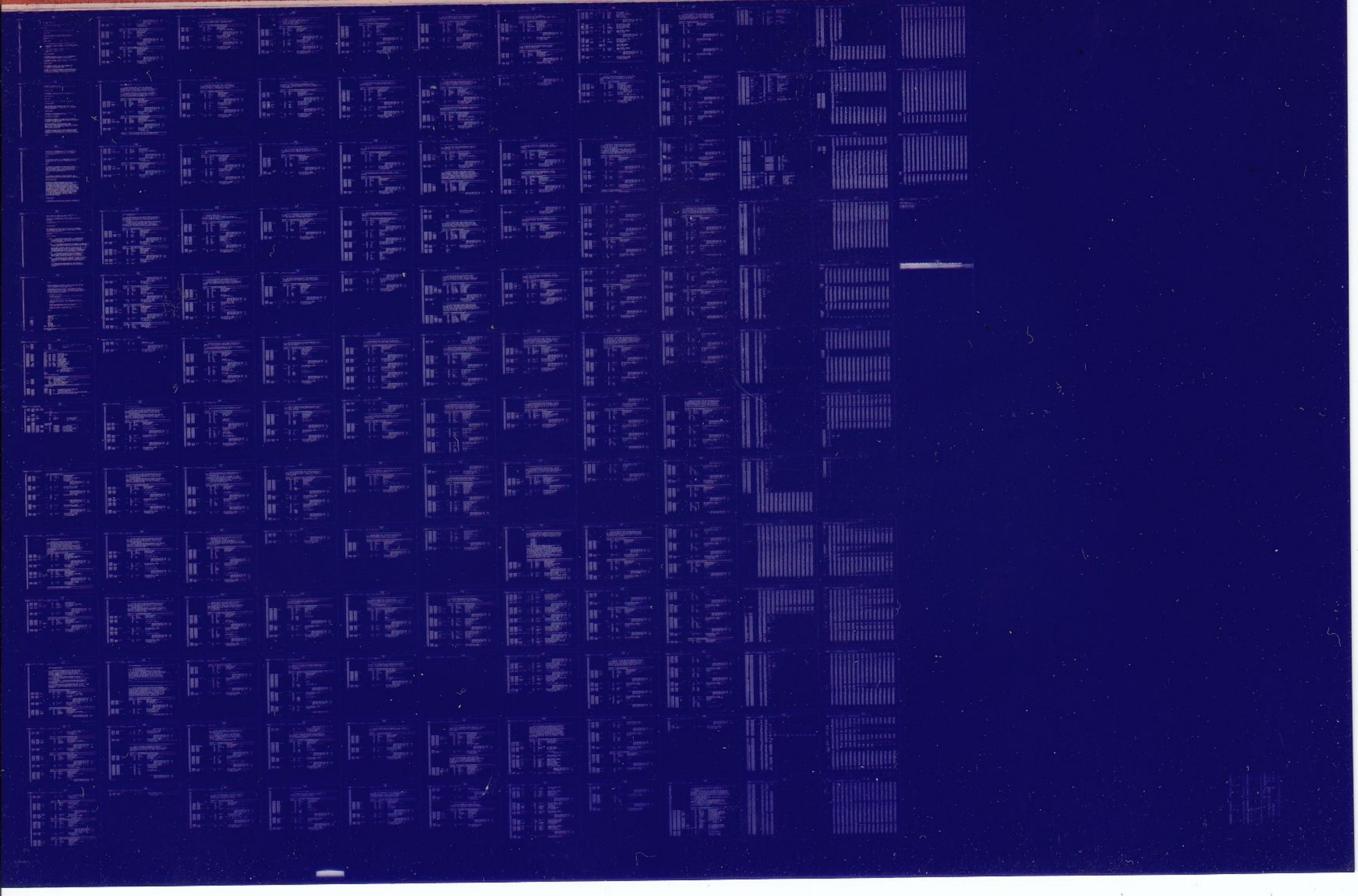
EP-DGKAA-A-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN US



2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURES

2.1.1 LOADING

USE NORMAL PROCEDURES FOR LOADING ABSOLUTE BINARY TAPES.

2.1.2 NORMAL START

THIS IS THE PROCEDURE FOR NORMAL PROGRAM RUNNING (I.E., STARTING WITH TEST 1 AND EXECUTING ENTIRE DIAGNOSTIC).

LOAD ADDRESS = 200
START

2.1.3 SUBTEST START

THIS IS THE PROCEDURE FOR STARTING AT A SUBTEST OTHER THAN 1.

1. LOAD $TESTN (IN MAILBOX SECTION) WITH THE NUMBER OF SUBTEST MINUS ONE (IN OCTAL) FOR EXAMPLE, TO START AT SUBTEST 10C, $TESTN=77.

2. LOAD STARTING ADDRESS OF SUBTEST IN LOC. 216

3. LOAD ADDRESS = ADDRESS OF SUBTEST
   START

2.2 SPECIAL ENVIRONMENTS

THIS PROGRAM IS WRITTEN TO COMPLY WITH ALL THE REQUIREMENTS OF THE APT INTERFACE SPECIFICATION. IT WILL RUN UNDER APT IN EITHER QUICK VERIFY, PROGRAM OR RUN-TIME MODES.

THIS PROGRAM IS WRITTEN TO COMPLY WITH ALL OF THE REQUIREMENTS OF PROGRAMS TO RUN UNDER THE ACT11 MONITOR.

2.3 PROGRAM OPTIONS

THIS PROGRAM IS INTENDED TO BE A BASIC PROCESSOR TEST.
IT IS INTENDED TO BE THE LOWEST LEVEL DIAGNOSTIC RUN.
IT PROVIDES FOR NO SELECTABLE OPTIONS.

IN ORDER THAT THE TEST BE RUNNABLE ON A PROCESSOR WITHOUT A TELETYPE, IT IS POSSIBLE TO SUPPRESS THE END OF PASS MESSAGE. IF NO TELETYPE IS AVAILABLE, ALTER THE BYTE, $ENVM, WHICH IS LOCATED IN THE APT MAILBOX. SETTING $ENVM TO 40(8) WILL

SUPPRESS ALL CONSOLE OUTPUT.
THE EXACT LOCATION OF THIS BYTE CAN BE FOUND IN THE SYMBOL TABLE AT THE END OF THE LISTING.

2.4 EXECUTION TIMES

THE DIAGNOSTIC COMPLETES ONE PASS IN LESS THAN 1 SEC.
THE PROGRAM WILL RUN CONTINUOUSLY UNTIL EXTERNALLY HALTED.

3.0 ERROR INFORMATION

3.1 ERROR TYPES

THERE ARE TWO BASIC TYPES OF ERRORS IN THE DIAGNOSTIC.

3.1.1 FUNCTIONAL ERRORS

THESE ARE ERRORS WHICH REPRESENT A MALFUNCTION OF AN INSTRUCTION OR SEQUENCE OF INSTRUCTION. (E.G., THE PROPER CONDITION CODE NOT SET OR IMPROPER RESULT OF AN ARITHMETIC OR LOGICAL OPERATION).

3.1.2 SEQUENCE ERRORS

THE RESULT OF A TESTS BEING EXECUTED OUT OF SEQUENCE. (E.G. WILD MACHINE OR IMPROPER BRANCH OR JUMP).

3.2 ERROR REPORTING PROCEDURES

THE DIAGNOSTIC RESPONDS TO THE DETECTION OF ALL ERRORS BY STORING CERTAIN INFORMATION IN MEMORY AND HALTING THE PROCESSOR. THE INFORMATION STORED IN MEMORY CAN BE USED BY THE OPERATOR TO IDENTIFY THE ERROR DETECTED.

CERTAIN FAILURES WILL CAUSE THE PROESSOR TO HANG.
THIS TYPE OF FAILURE IS INDICATED IF THE PROGRAM
DOES NOT PRINT ITS END OF PASS INDICATION WITHIN A REASONABLE
AMOUNT OF TIME. (FIRST MESSAGE SHOULD APPEAR WITHIN 1 SEC.)

3.3 ERROR DESCRIPTOR INFORMATION

THE DIAGNOSTIC MAILBOX HOLDS THE ERROR INFORMATION NECESSARY TO IDENTIFY THE DETECTED ERROR. THIS INFORMATION HAS BEEN DESIGNED FOR COMPLIANCE WITH THE APT TO DIAGNOSTIC INTERFACE SPECIFICATION. IT IS THE PRIMARY MEDIUM FOR IDENTIFYING ERRORS.

3.2.1 $MSGTYP

THIS LOCATION IS INCREMENTED FROM ZERO TO ONE BEFORE THE PROGRAM COMES TO A PROGRAMMED HALT. IF THIS LOCATION IS NOT ONE, THEN THE DIAGNOSTIC HAS COME TO AN UNPROGRAMMED HALT. CHECK THE STACK AND PC FOR A CLUE TO THE CAUSE. SUSPECT A TRAP.

3.2.2 $FATAL

THIS LOCATION IS LOADED WITH A NUMBER BEFORE A HALT IS EXECUTED. EACH PROGRAMMED HALT HAS A UNIQUE NUMBER ASSOCIATED WITH IT WHICH CAN BE USED TO IDENTIFY THE ERROR WHICH HAS BEEN DETECTED.

3.2.3 $PASS

THIS LOCATION IS INCREMENTED FOR EVERY COMPLETE PASS OF THE DIAGNOSTIC. MONITORING THIS LOCATION WILL INDICATE WHETHER OR NOT THE PROGRAM IS HUNG. IT WILL ALSO INDICATE THE NUMBER OF SUCCESSFUL PASSES COMPLETED BEFORE THE ERROR HALT. A HIGH PASS COUNT MIGHT INDICATE THAT THE ERROR HALT IS ASSOCIATED WITH AN INTERMITTANT FAULT.

3.2.4 $TESTN

THIS LOCATION IS INCREMENTED IN EACH NEW SUBTEST. THIS SHOULD INDICATE THE TEST BEING EXECUTED WHEN THE ERROR WAS DETECTED. THIS LOCATION IS ALSO USED TO DETECT A SEQUENCE ERROR.

3.4 ERROR IDENTIFICATION

BECAUSE OF THE OVERHEAD ASSOCIATED WITH EACH HALT IN AN APT COMPATIBLE PROGRAM THE SEQUENCE CHECK CODE WILL SHARE THE ERROR HALT OF FUNCTIONAL ERROR WITHIN EACH SUBTEST.
TO DETERMINE WHICH ERROR IS BEING REPORTED, LOCATIONS $FATAL AND $TESTN ARE USED TOGETHER. WHEN AN ERROR HALT OCCURS, CHECK, $FATAL TO DETERMINE THE NUMBER OF THE ERROR DETECTED. NOW, CHECK THAT THE TEST NUMBER WHERE THIS ERROR IS DETECTED CORRESPONDS TO THE VALUE IN $TESTN. IF THESE AGREE THE ERROR WAS A FUNCTIONAL ERROR AS DESCRIBED IN THE LISTINGS. IF THESE NUMBERS DO NOT AGREE, THEN A SEQUENCE ERROR WAS DETECTED. IN THIS CASE $TESTN WILL CONTAIN ONE MORE THAN THE NUMBER OF THE LAST TEST SUCCESSFULLY COMPLETED. SEQUENCE ERRORS WHICH SHARE THE ERROR HALTS OF FUNCTIONAL ERRORS WILL ALWAYS BE REPORTED BY THE LAST HALT IN THE SUBTEST IN WHICH THEY WERE DISCOVERED.

4.0 PROGRESS REPORT

AT THE END OF EACH SUCCESSFUL PASS THE PROGRAM INCREMENTS THE LOCATION $PASS WHICH IS IN THE APT MAILBOX. THIS LOCATION WIL

ALWAYS CONTAIN THE NUMBER OF SUCCESSFUL PASSES COMPLETED.
$PASS IS RESET WITH EVERY RETART FROM LOC. 200.

ADDITIONALLY, THE MESSAGE END OF DGKAA IS PRINTED ON THE CONSOLE TELETYPE AFTER THE FIRST PASS AND FOLLOWING EVERY 400TH PASS THEREAFTER.

IF NO TELETYPE IS AVAILABLE, THE CONSOLE OUTPUT MUST BE SUPPRESSED. (SEE SECTION 2.3)

5.0 TROUBLE SHOOTING

WHEN THE PROGRAM DISCOVERS A FAULT IT WILL HALT. TO DETERMINE THE CAUSE OF THE HALT, THE DIAGNOSTIC PROVIDES ERROR INFORMATION. THIS INFORMATION IS STORED IN THE APT MAILBOX AND IS THE PRIMARY SOURCE OF ERROR IDENTIFICATION.

UPON FINDING AN ERROR, THE FOLLOWING PROCEDURE SHOULD AID IN ISOLATING THE FAULT.

5.1 CHECK THE MAILBOX

1. $MSGTY — THIS LOCATION SHOULD CONTAIN A 1. IF THE PROCESSOR HALTS AND THIS LOCATION IS ZERO, THEN THE PROCESSOR HAS COME TO AN UNEXPECTED HALT. FIRST SUSPECT A TRAP. CHECK THE PC AND IF A TRAP CHECK R6 AND THE STACK FOR THE LOCATION OF THE FAILING INSTRUCTION.

2. $FATAL — THIS LOCATION IS USED TO HOLD THE NUMBER OF THE ERROR WHICH HAS DETECTED. EACH ERROR BEING CHECKED BY THE DIAGNOSTIC IS ASSIGNED A UNIQUE NUMBER WHICH IS STORED IN $FATAL WHEN THAT ERROR IS DETECTED.

   WHEN AN ERROR IS DETECTED, CHECK THE LISTING TO SEE THAT THE ERROR NUMBER STORED IN $FATAL IS ONE WHICH IS DETECTED IN THE TEST WHOSE NUMBER IS IN $TESTN. IF THERE IS A DISAGREEMENT THEN THE ERROR BEING REPORTED IS A SEQUENCE ERROR. $TESTN CONTAINS ONE MORE THAN THE LAST TEST WHICH WAS SUCCESSFULLY COMPLETED.

3. $TESTN — THIS LOCATION IS USED TO INDICATE THE NUMBER OF THE TEST WHICH WAS BEING EXECUTED WHEN THE FUALT WAS DETECTED. $TESTN IS USED IN CONJUCTION WITH $FATAL TO DISTINGUISH BETWEEN SEQUENCE AND FUNCTONAL ERRORS. (SEE 2. THIS SECTION)

4. $PASS — THIS LOCATION IS USED TO INDICATE THE NUMBER OF SUCCESSFUL PASSES WHICH THE DIAGNOSTIC HAS COMPLETED. THIS WILL GIVE AN INDICATION THAT THE DIAGNOSTIC HAS NOT JUST BEEN HUNG IN A LOOP IF NOT TELETYPE IS AVAILABLE TO REPORT THE PRINTED PROGRESS REPORTS.

   IF AN ERROR HAS BEEN DETECTED $PASS WILL SHOW WHETHER IT WAS A HARD ERROR DISCOVERED DURING THE FIRST TRY OR WHETHER IT WAS INTERMITTANT OR DEVERLOPED DURING THE RUNNING OF THE DIAGNOSTIC.

```
                                5.2     SCOPING
                                        -------

                                WHILE THIS DIAGNOSTIC IS PRIMARILY INTENDED TO BE A FAULT DETECTION
                                PROGRAM, PROVISIONS ARE MADE TO ASSIST A TECHNICIAN WHO MIGHT WANT
                                TO USE THE PROGRAM AS A TROUBLE SHOOTING TEST.

                                THE PROCEDURE FOR SCOPING A SUBTEST INVOLVES MODIFYING SEVERAL
                                MEMORY LOCATIONS IN THE TEST ITSELF.  THE PHILOSOPHY IS TO PROVIDE
                                A SCOPING LOOP WHICH WILL INCLUDE THE CODE WHERE THE ERROR WAS DETECTED.
                                THE LOOP IS SET UP SO THAT THE LOOP WILL NOT BE TERMINATED SHOULD
                                THE ERROR INTERMITTANTLY DISAPPEAR.

                                THE PROCEDURE IS AS FOLLOWS:

                                1.  DETERMINE WHICH ERROR IS TO BE SCOPED.  USE $FATAL AND $TESTN
                                    FOR THIS (SEE ABOVE)

                                2.  LOCATE THE ERROR ROUTINE IN THE LISTING.

                                3.  CLEAR THE RIGHT BYTE OF THE CONDITIONAL BRANCH INSTRUCTION
                                    ASSOCIATED WITH THE ERROR.  (THIS IS MARKED WITH <===='S IN THE
                                    LISTING.)

                                4.  REPLACE THE INSTRUCTION FOLLOWING <MOV #XXX,-(R2)) WITH THE
                                    SCOPING BRANCH PROVIDED IN THE LISTING COMMENTS.

                                5.  RESTART THE PROGRAM.  THE PROGRAM MAY BE RESTARTED FROM THE
                                    BEGINNING OR FROM THE SUBTEST (SEE 2.0).

                        6.0     LISTING
                                -------


                                %
                                .TITLE  MAINDEC-11-DGKAA  11/04 CPU TEST
                                .ENABLE ABS
000500                          STBOT=500
                                .NLIST  CND,MC,MD
                                .LIST   ME
000240                          SCOPE=NOP
000007                          R7=%7
000006                          R6=%6
177776                          PS=177776
177564                          TPS=177564
177566                          TPB=177566
                        .MCALL  .$APTHDR,.$APTBLS,.$ACT11
                        .SBTTL  ACT11 HOOKS

                        ;*****************************************************************************
                        ;HOOKS REQUIRED BY ACT11
000400C                         $SVPC=.                 ;SAVE PC
```

```
                    000046
000046  015264
                    000052
000052  000000
                    000400
                    000300
                                .=46
                                $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN  SE P
                                .=52
                                .WORD   0               ;;2)SET LOC.52 TO ZERO
                                .=$SVPC                 ;; RESTORE PC
                                .=300
                            .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
                            .SBTTL  APT PARAMETER BLOCK

                            ;*****************************************************************************
                            ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
                            ;*****************************************************************************
                                    .$X=.       ;;SAVE CURRENT LOCATION
                                    .=24        ;;SET POWER FAIL TO POINT TO START OF PROGRAM
                                    200         ;;FOR APT START UP
                                    .=44        ;;POINT TO APT INDIRECT ADDRESS PNTR.
                                    $APTHDR     ;;POINT TO APT HEADER BLOCK
                                    .=.$X       ;;RESET LOCATION COUNTER
                            ;*****************************************************************************
                            ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                            ;INTERFACE SPEC.

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  000002             $TSTM:  .WORD   2           ;;RUN TIM OF LONGEST TEST
000336  000002             $PASTM: .WORD   2           ;;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)
                    000370          .=370
```

```
000370  000000  000000  000000          0,0,0,0,0,0
000376  000000  000000  000000
000404  000001  000001  177777          1,1,-1
        000500                          .=500
                                ;*****************************************************************************
                                ;SET UP STARTING ADDRESS
        000500                  $X=.
        000200                  .=200
000200  000167  000274          JMP     START

000204  012706  000500          MOV     #STBOT,R6               ;SET STACK POINTER
000210  012702  000304          MOV     #$TESTN,R2              ;SET MAILBOX POINTER
000214  000137                  JMP     @(PC)+                  ;JUMP TO SUBTEST
000216  000000                  0                               ;ADDR. OF SUBTEST GOES HERE

        000500                  .=.$X
        000302          $ERROR=$FATAL
        000304          $TSTNM=$TESTN
000500  012737  015426  000024  START:  MOV     #PWRDN,@#24             ;SET UP FOR POWER FAIL
000506  012737  000000  000306          MOV     #0,@#$PASS      ;CLEAR PASS COUNT
000514  012737  177777  015310          MOV     #-1,@#PASSPT    ;SET PRINT COUNTER
000522  012706  000500          RESTRT: MOV     #STBOT,R6       ;INITIALIZE STACK POINTER
000526  012702  000304                  MOV     #$TESTN,R2      ;SET UP POINTER TO MESSAGE TYPE
000532  012737  000000  000304          MOV     #0,@#$TSTNM     ;CLEAR TEST NUMBER
000540  012737  000000  000302          MOV     #0,@#$ERROR     ;CLEAR ERROR NUMBER
000546  012737  000000  000300          MOV     #0,@#$MSGTY     ;CLEAR MESSAGE TYPE(FOR APT)
```

```
                                ;*****************************************************************
                                ;TEST 1 CHECK BRANCHES ON Z BIT
                                ;*****************************************************************
000554  005212                  TST1:   INC     (R2)            ;UPDATE TEST NUMBER
000556  022712  000001                  CMP     #1,(R2)         ;SEQUENCE ERROR?
000562  001024                          BNE     TST2-10         ;BR TO ERROR HALT ON SEQ ERROR
000564  000257                          CCC                     ;CLEAR ALL CONDITION CODES
000566  001401                          BEQ     BR1             ;SHOULD BRANCH
000570  000404                          BR      BR2             ;BAD BRANCH OF Z-BIT
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                                ;           BRANCH INSTRUCTION AND            <====
                                                                ;           REPLACE THE MOVE INSTRUCTION      <====
                                                                ;           FOLLOWING W/ 774                  ====
000572                          BR1:
000572  012742  000001                  MOV     #1,-(R2)        ;MOVE TO MAILBOX #  *******  1  *******
000576  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
000600  000000                          HALT                    ;SHOULD HAVE BRANCHED: Z=0
000602                          BR2:
000602  001004                          BNE     BR3
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                                ;           CONDITIONAL BRANCH INST. AND      <====
                                                                ;           REPLACE THE MOVE INSTRUCTION      <====
                                                                ;           WHICH FOLLOWS W/ 770              <====
000604  012742  000002                  MOV     #2,-(R2)        ;MOVE TO MAILBOX #  *******  2  *******
000610  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
000612  000000                          HALT                    ;
000614  000264                  BR3:    SEZ
000616  001001                          BNE     BR4
000620  000404                          BR      BR5
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                                ;           BRANCH INSTRUCTION AND            <====
                                                                ;           REPLACE THE MOVE INSTRUCTION      <====
                                                                ;           FOLLOWING W/ 760                  <====
000622                          BR4:
000622  012742  000003                  MOV     #3,-(R2)        ;MOVE TO MAILBOX #  *******  3  *******
000626  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
000630  000000                          HALT                    ;SHOULD NOT HAVE BRANCHED HERE ON Z=1
000632                          BR5:
000632  001404                          BEQ     TST2
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                                ;           CONDITIONAL BRANCH INST. AND      <====
                                                                ;           REPLACE THE MOVE INSTRUCTION      <====
                                                                ;           WHICH FOLLOWS W/ 754              <====
000634  012742  000004                  MOV     #4,-(R2)        ;MOVE TO MAILBOX #  *******  4  *******
000640  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
000642  000000                          HALT                    ;SHOULD HAVE BRANCHED ON Z=1
                                                                ;  OR SEQUENCE ERROR
```

```
                                        ;*****************************************************************************
                                        .SBTTL  DATA PATH TESTS
                                        ;
                                        ;       THE DATA PATH TESTS ARE USED TO VERIFY THAT VARIOUS
                                        ;DATA PATTERNS CAN BE SUCCESSFULLY MOVED THROUGH THE DATA PATHS
                                        ;MOVE AND COMPARE MODE 2,3 INSTRUCTIONS ARE USED TO PASS AND
                                        ;TEST VARIOUS DATA PATTERNS IN THE DATA PATHS.
                                        ;       THE TEST EXERCISES THE INTERNAL DATA PATHS, THE UNIBUS
                                        ;DATA TRANSCIEVERS, AND AMUX CONTROL FOR ALU AND UBUS INPUTS.
                                        ;       IF THESE TESTS FAIL, EXAMINE THE TARGET LOCATION (LOC. 0)
                                        ;TO SEE WHICH BITS OF THE DATA PATH ARE FAILING.  IF THIS PROVIDES
                                        ;INCONCLUSIVE DATA, TRY TO CHECK MODE 3 IR DECODE BY RUNNING
                                        ;JUST THE MICROCODE AND IR DECODE TESTS FOR THE MOVE AND COMPARE
                                        ;INSTRUCTIONS.
                                        ;*****************************************************************************
                                        ;TEST 2 TEST OF ZEROES IN THE DATA PATH
                                        ;*****************************************************************************
000644  005212                  TST2:   INC     (R2)            ;UPDATE TEST NUMBER
000646  022712  000002                  CMP     #2,(R2)         ;SEQUENCE ERROR?
000652  001006                          BNE     TST3-10         ;BR TO ERROR HALT ON SEQ ERROR
000654  012737  000000  000000          MOV     #0,@#0          ;MOVE ZEROES THRU ADDRESS LINES, DATA
                                                                ;LINES AND INTERNAL PATHS
000662  005737  000000                  TST     @#0             ;SUCCESSFUL?
000666  001404                          BEQ     TST3
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 772           <====
000670  012742  000005                  MOV     #5,-(R2)        ;MOVE TO MAILBOX #  *******  5  *******
000674  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
000676  000000                          HALT                    ;DATA INCORRECT
                                                                ;  OR SEQUENCE ERROR

                                        ;*****************************************************************************
                                        ;TEST 3 TEST OF PATTERN 125252 IN DATA PATH
                                        ;*****************************************************************************
000700  005212                  TST3:   INC     (R2)            ;UPDATE TEST NUMBER
000702  022712  000003                  CMP     #3,(R2)         ;SEQUENCE ERROR?
000706  001007                          BNE     TST4-10         ;BR TO ERROR HALT ON SEQ ERROR
000710  012737  125252  000000          MOV     #125252,@#0     ;MOVE ALTERNATING ONES AND ZEROES
                                                                ;THRU DATA PATHS
000716  022737  125252  000000          CMP     #125252,@#0     ;SUCCESSFUL
000724  001404                          BEQ     TST4
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 771           <====
000726  012742  000006                  MOV     #6,-(R2)        ;MOVE TO MAILBOX #  *******  6  *******
000732  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
000734  000000                          HALT                    ;DATA INCORRECT
                                                                ;  OR SEQUENCE ERROR

                                        ;*****************************************************************************
                                        ;TEST 4 TEST OF PATTERN 052525 IN DATA PATH
                                        ;*****************************************************************************
```

```
                                TST4:   INC     (R2)            ;UPDATE TEST NUMBER
                                        CMP     #4,(R2)         ;SEQUENCE ERROR?
                                        BNE     TST5-10         ;BR TO ERROR HALT ON SEQ ERROR
                                        MOV     #052525,@#0     ;MOVE ALTERNATING ZEROES AND ONES
                                                                ;THRU DATA PATH
                                        CMP     #052525,@#0     ;SUCCESSFUL?
                                        BEQ     TST5
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 771           <====
                                        MOV     #7,-(R2)        ;MOVE TO MAILBOX #  *******  7  *******
                                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
                                        HALT                    ;DATA INCORRECT
                                                                ;  OR SEQUENCE ERROR

                                ;***************************************************************************
                                ;TEST 5 TEST OF ALL ONES IN DATA PATH
                                ;***************************************************************************
                                TST5:   INC     (R2)            ;UPDATE TEST NUMBER
                                        CMP     #5,(R2)         ;SEQUENCE ERROR?
                                        BNE     TST6-10         ;BR TO ERROR HALT ON SEQ ERROR
                                        MOV     #177777,@#0     ;MOVE ONES THRU DATA PATH
                                        CMP     #177777,@#0     ;SUCCESSFUL
                                        BEQ     TST6
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 771           <====
                                        MOV     #10,-(R2)       ;MOVE TO MAILBOX #  *******  10  *******
                                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
                                        HALT                    ;DATA INCORRECT
                                                                ;  OR SEQUENCE ERROR
```

```
                        ;*****************************************************************************
                        .SBTTL  SCRATCH PAD TESTS
                        ;
                        ;       THE SCRATCH PAD TESTS ARE USED TO VERIFY THAT VARIOUS
                        ;DATA PATTERNS CAN BE SUCCESSFULLY HELD IN THE SCRATCH PAD
                        ;CIRCUITRY.  MOVE AND COMPARE INSTRUCTIONS ARE USED TO TEST THAT
                        ;R0 CAN HOLD VARIOUS DATA PATTERNS.  EACH DATA PATTERN IS
                        ;MOVED AND TESTED IN A SMALL LOOP CONVENIENT FOR SCOPING.  THE
                        ;SUCCESSFUL COMPLETION OF THESE TESTS SHOULD VERIFY THE CIRCUITRY EXTERNAL
                        ;TO THE SCRATCH PAD ITSELF.
                        ;       THE REMAINDER OF THE GENERAL REGISTERS ARE TESTED BY MOVING
                        ;A BIT INTO BIT 0 OF THE REGISTER AND SHIFTING IT LEFT ONE
                        ;BIT AT A TIME INTO THE CARRY BIT.  THE RESULT IS THEN CHECKED TO INSURE THAT
                        ;NO BITS WERE PICKED.
                        ;       AT THIS POINT ALL OF THE GENERAL REGISTERS HAVE BEEN EXERCISED
                        ;AS WELL AS REGISTER 11.  REGISTERS 10 AND 12 HAVE BEEN ACCESSED BY
                        ;THE INSTRUCTIONS.  REGISTERS 13,14,AND 17 WILL BE TESTED LATER IN THE
                        ;MICROCODE TESTS.
                        ;       IF THE PATTERN TESTS WITH REGISTER 0 FAIL CHECK THE RESULTANT
                        ;DATA FOR A CLUE TO A FAULT IN THE EXTERNAL CIRCUITRY.  IF THE
                        ;PATTERN TESTS WITH R0 ARE SUCCESSFUL BUT THE TESTS WITH THE OTHER
                        ;REGISTERS FAIL, SUSPECT THE REGISTER SELECT LINES AND THEN THE SCRATCH
                        ;PAD ITSELF.
                        ;
                        ;*****************************************************************************
                        ;TEST 6 TEST IF R0 CAN HOLD ALL ZEROES
                        ;*****************************************************************************
001032  005212          TST6:   INC     (R2)            ;UPDATE TEST NUMBER
001034  022712  000006          CMP     #6,(R2)         ;SEQUENCE ERROR?
001040  001004                  BNE     TST7-10         ;BR TO ERROR HALT ON SEQ ERROR
                        ;
001042  012700  000000          MOV     #0,R0           ;MOVE ZEROES TO R0
001046  005700                  TST     R0              ;SUCCESSFUL?
001050  001404                  BEQ     TST7
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 774           <====
001052  012742  000011          MOV     #11,-(R2)       ;MOVE TO MAILBOX #  *******  11  *******
001056  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001060  000000                  HALT                    ;R0 NOT 0
                                                        ;  OR SEQUENCE ERROR
                        ;
                        ;*****************************************************************************
                        ;TEST 7 TEST IF R0 CAN HOLD ONES AND ZEROES
                        ;*****************************************************************************
001062  005212          TST7:   INC     (R2)            ;UPDATE TEST NUMBER
001064  022712  000007          CMP     #7,(R2)         ;SEQUENCE ERROR?
001070  001005                  BNE     TST10-10        ;BR TO ERROR HALT ON SEQ ERROR
001072  012700  125252          MOV     #125252,R0      ;MOVE ALTERNATING ONES AND ZEROES TO R0
001076  020027  125252          CMP     R0,#125252      ;SUCCESSFUL?
001102  001404                  BEQ     TST10
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
```

```
                                                ;          WHICH FOLLOWS W/ 773
001104  012742  000012                MOV     #12,-(R2)   ;MOVE TO MAILBOX #  *******  12  *******
001110  005242                        INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
001112  000000                        HALT                ;R0 NOT 125252
                                                ;  OR SEQUENCE ERROR


                        ;*****************************************************************************
                        ;TEST 10        TEST IF R0 CAN HOLD ZEROES AND ONES
                        ;*****************************************************************************
001114  005212          TST10:  INC     (R2)            ;UPDATE TEST NUMBER
001116  022712  000010          CMP     #10,(R2)        ;SEQUENCE ERROR?
001122  001005                  BNE     TST11-10        ;BR TO ERROR HALT ON SEQ ERROR
001124  012700  052525          MOV     #052525,R0      ;MOVE ALTERNATING ZEROES AND ONES TO R0
001130  020027  052525          CMP     R0,#052525      ;SUCCESSFUL?
001134  001404                  BEQ     TST11
                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                ;           WHICH FOLLOWS W/ 773           <====
001136  012742  000013                MOV     #13,-(R2)   ;MOVE TO MAILBOX #  *******  13  *******
001142  005242                        INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
001144  000000                        HALT                ;R0 NOT 52525
                                                ;  OR SEQUENCE ERROR


                        ;*****************************************************************************
                        ;TEST 11        TEST IF R0 CAN HOLD ALL ONES
                        ;*****************************************************************************
001146  005212          TST11:  INC     (R2)            ;UPDATE TEST NUMBER
001150  022712  000011          CMP     #11,(R2)        ;SEQUENCE ERROR?
001154  001005                  BNE     TST12-10        ;BR TO ERROR HALT ON SEQ ERROR
001156  012700  177777          MOV     #177777,R0      ;MOVE ALL ONES TO R0
001162  020027  177777          CMP     R0,#177777      ;SUCCESSFUL?
001166  001404                  BEQ     TST12
                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                ;           WHICH FOLLOWS W/ 773           <====
001170  012742  000014                MOV     #14,-(R2)   ;MOVE TO MAILBOX #  *******  14  *******
001174  005242                        INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
001176  000000                        HALT                ;R0 NOT 177777
                                                ;  OR SEQUENCE ERROR


                        ;*****************************************************************************
                        ;TEST 12        TEST IF R1 CAN HOLD A ONE IN ALL BITS
                        ;*****************************************************************************
001200  005212          TST12:  INC     (R2)            ;UPDATE TEST NUMBER
001202  022712  000012          CMP     #12,(R2)        ;SEQUENCE ERROR?
001206  001006                  BNE     TST13-10        ;BR TO ERROR HALT ON SEQ ERROR
001210  012701  000001          MOV     #1,R1           ;SET BIT 0
001214  000241                  CLC                     ;CLEAR C-BIT
001216  006101          REG1:   ROL     R1              ;ROTATE 1 POSITION
001220  103376                  BCC     REG1            ;ALL DONE
001222  001404                  BEQ     TST13
                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                ;           REPLACE THE MOVE INSTRUCTION   <====
```

```
                                                ;            WHICH FOLLOWS W/ 772          <====
001224  012742  000015                  MOV   #15,-(R2)     ;MOVE TO MAILBOX #  *******  15  *******
001230  005242                          INC   -(R2)         ;SET MSGTYP TO FATAL ERROR
001232  000000                          HALT                ;FAILURE WITH R1
                                                            ;  OR SEQUENCE ERROR


                                ;***************************************************************************
                                ;TEST 13        TEST IF R2 CAN HOLD A ONE IN ALL BITS
                                ;***************************************************************************
001234  005212                  TST13:  INC   (R2)          ;UPDATE TEST NUMBER
001236  022712  000013                  CMP   #13,(R2)      ;SEQUENCE ERROR?
001242  001006                          BNE   REG2A-14              ;BR TO ERROR HALT ON SEQ ERROR
001244  012702  000001                  MOV   #1,R2         ;SET BIT 0
001250  000241                          CLC                 ;CLEAR C-BIT
001252  006102                  REG2:   ROL   R2            ;ROTATE 1 POSITION
001254  103376                          BCC   REG2          ;ALL DONE
001256  001406                          BEQ   REG2A
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           BRANCH INSTRUCTION AND        <====
                                                            ;           REPLACE THE MOVE INSTRUCTION  <====
                                                            ;           FOLLOWING W/ 771              <====
001260  012702  000304                  MOV   #$TESTN,R2    ;RESTORE POINTER
001264  012742  000016                  MOV   #16,-(R2)     ;MOVE TO MAILBOX #  *******  16  *******
001270  005242                          INC   -(R2)         ;SET MSGTYP TO FATAL ERROR
001272  000000                          HALT                ;FAILURE WITH R2
001274  012702  000304          REG2A:  MOV   #$TESTN,R2            ;RESTORE POINTER
                                ;***************************************************************************
                                ;TEST 14        TEST IF R3 CAN HOLD A ONE IN ALL BITS
                                ;***************************************************************************
001300  005212                  TST14:  INC   (R2)          ;UPDATE TEST NUMBER
001302  022712  000014                  CMP   #14,(R2)      ;SEQUENCE ERROR?
001306  001006                          BNE   TST15-10      ;BR TO ERROR HALT ON SEQ ERROR
001310  012703  000001                  MOV   #1,R3         ;SET BIT 0
001314  000241                          CLC                 ;CLEAR C-BIT
001316  006103                  REG3:   ROL   R3            ;ROTATE 1 POSITION
001320  103376                          BCC   REG3          ;ALL DONE
001322  001404                          BEQ   TST15
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND  <====
                                                            ;           REPLACE THE MOVE INSTRUCTION  <====
                                                            ;           WHICH FOLLOWS W/ 772          <====
001324  012742  000017                  MOV   #17,-(R2)     ;MOVE TO MAILBOX #  *******  17  *******
001330  005242                          INC   -(R2)         ;SET MSGTYP TO FATAL ERROR
001332  000000                          HALT                ;FAILURE WITH R3
                                                            ;  OR SEQUENCE ERROR


                                ;***************************************************************************
                                ;TEST 15        TEST IF R4 CAN HOLD A ONE IN ALL BITS
                                ;***************************************************************************
001334  005212                  TST15:  INC   (R2)          ;UPDATE TEST NUMBER
001336  022712  000015                  CMP   #15,(R2)      ;SEQUENCE ERROR?
001342  001006                          BNE   TST16-10      ;BR TO ERROR HALT ON SEQ ERROR
001344  012704  000001                  MOV   #1,R4         ;SET BIT 0
001350  000241                          CLC                 ;CLEAR C-BIT
001352  006104                  REG4:   ROL   R4            ;ROTATE 1 POSITION
001354  103376                          BCC   REG4          ;ALL DONE
```

```
001356  001404                          BEQ     TST16
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 772           <====
001360  012742  000020                  MOV     #20,-(R2)       ;MOVE TO MAILBOX #  *******  20  *******
001364  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001366  000000                          HALT                    ;FAILURE WITH R4
                                                                ;  OR SEQUENCE ERROR

                                ;*****************************************************************************
                                ;TEST 16         TEST IF R5 CAN HOLD A ONE IN ALL BITS
                                ;*****************************************************************************
001370  005212                  TST16:  INC     (R2)            ;UPDATE TEST NUMBER
001372  022712  000016                  CMP     #16,(R2)        ;SEQUENCE ERROR?
001376  001006                          BNE     TST17-10        ;BR TO ERROR HALT ON SEQ ERROR
001400  012705  000001                  MOV     #1,R5           ;SET BIT 0
001404  000241                          CLC                     ;CLEAR C-BIT
001406  006105                  REG5:   ROL     R5              ;ROTATE 1 POSITION
001410  103376                          BCC     REG5            ;ALL DONE
001412  001404                          BEQ     TST17
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 772           <====
001414  012742  000021                  MOV     #21,-(R2)       ;MOVE TO MAILBOX #  *******  21  *******
001420  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001422  000000                          HALT                    ;FAILURE WITH R5
                                                                ;  OR SEQUENCE ERROR

                                ;*****************************************************************************
                                ;TEST 17         TEST IF R6 CAN HOLD A ONE IN ALL BITS
                                ;*****************************************************************************
001424  005212                  TST17:  INC     (R2)            ;UPDATE TEST NUMBER
001426  022712  000017                  CMP     #17,(R2)        ;SEQUENCE ERROR?
001432  001006                          BNE     TST20-10        ;BR TO ERROR HALT ON SEQ ERROR
001434  012706  000001                  MOV     #1,R6           ;SET BIT 0
001440  000241                          CLC                     ;CLEAR C-BIT
001442  006106                  REG6:   ROL     R6              ;ROTATE 1 POSITION
001444  103376                          BCC     REG6            ;ALL DONE
001446  001404                          BEQ     TST20
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 772           <====
001450  012742  000022                  MOV     #22,-(R2)       ;MOVE TO MAILBOX #  *******  22  *******
001454  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001456  000000                          HALT                    ;FAILURE WITH R6
                                                                ;  OR SEQUENCE ERROR
```

```
                                        ;*****************************************************************
                                        .SBTTL  PSW TESTS
                                        ;
                                        ;       THE PSW TESTS ARE USED TO VERIFY THAT VARIOUS DATA
                                        ;PATTERNS CAN BE SUCCESSFULLY HELD IN THE PSE AND THAT THE
                                        ;PSW ADDRESSING LOGIC IS FUNCTIONING.  MOVE AND COMPARE INSTRUCTIONS
                                        ;ARE USED TO TEST THAT THE PSE CAN HOLD VARIOUS DATA PATTERNS.
                                        ;EACH DATA PATTERN IS MOVED AND TESTED IN A SMALL LOOP CONVENIENT FOR
                                        ;SCOPING.
                                        ;       THE PSW REGISTER ITSELF IS TESTED AS WELL AS THE ADDRESS
                                        ;SELECT CIRCUITRY.  THE AMUX INPUTS TO THE PSW MUX ARE TESTED.  THE
                                        ;CC INPUTS ARE TESTED LATER IN THE MICROCODE TESTS.  SETTING OF
                                        ;THE T-BIT BY THE TEST PATTERNS IS PURPOSELY AVOIDED; TESTING OF THE
                                        ;T-BIT TRAP CIRCUITRY IS LEFT FOR THE TRAP TEST.
                                        ;
                                        ;*****************************************************************
                                        ;TEST 20        TEST IF PSW WILL HOLD ZEROES
                                        ;*****************************************************************
001460  005212                        TST20:  INC     (R2)            ;UPDATE TEST NUMBER
001462  022712  000020                        CMP     #20,(R2)        ;SEQUENCE ERROR?
001466  001010                                BNE     TST21-10        ;BR TO ERROR HALT ON SEQ ERROR
001470  012706  000500                        MOV     #STBOT,R6
001474  012737  000000  177776                MOV     #0,@#PS         ;SET PSW TO ZERO
001502  005737  177776                        TST     @#PS            ;SUCCESSFUL
001506  001404                                BEQ     TST21
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                        ;           WHICH FOLLOWS W/ 770            <====
001510  012742  000023                        MOV     #23,-(R2)       ;MOVE TO MAILBOX #  *******  23  *******
001514  005242                                INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001516  000000                                HALT                    ;PSW NOT 0
                                                                        ;  OR SEQUENCE ERROR

                                        ;*****************************************************************
                                        ;TEST 21        TEST IF PSW WILL HOLD ONES AND ZEROES
                                        ;*****************************************************************
001520  005212                        TST21:  INC     (R2)            ;UPDATE TEST NUMBER
001522  022712  000021                        CMP     #21,(R2)        ;SEQUENCE ERROR?
001526  001007                                BNE     TST22-10        ;BR TO ERROR HALT ON SEQ ERROR
001530  012737  000252  177776                MOV     #252,@#PS       ;MOVE ALT. ONES AND ZEROES TO PSW
001536  023727  177776  000252                CMP     @#PS,#252       ;SUCCESSFUL?
001544  001404                                BEQ     TST22
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                        ;           WHICH FOLLOWS W/ 771            <====
001546  012742  000024                        MOV     #24,-(R2)       ;MOVE TO MAILBOX #  *******  24  *******
001552  005242                                INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001554  000000                                HALT                    ;PSW NOT 252
                                                                        ;  OR SEQUENCE ERROR

                                        ;*****************************************************************
                                        ;TEST 22        TEST IF PSW (EXCEPT T-BIT) WILL HOLD ZEROES AND ONES
                                        ;*****************************************************************
```

```
 907  001556  005212                  TST22:  INC     (R2)            ;UPDATE TEST NUMBER
 908  001560  022712  000022                  CMP     #22,(R2)        ;SEQUENCE ERROR?
 909  001564  001007                          BNE     TST23-10        ;BR TO ERROR HALT ON SEQ ERROR
 910  001566  012737  000105  177776          MOV     #105,@#PS       ;MOVE ALT. ONES AND ZEROES TO PSW
 911  001574  023727  177776  000105          CMP     @#PS,#105       ;SUCCESSFUL?
 912  001602  001404                          BEQ     TST23
 913                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 914                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
 915                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
 916                                                                  ;           WHICH FOLLOWS W/ 771           <====
 917  001604  012742  000025                  MOV     #25,-(R2)       ;MOVE TO MAILBOX #  *******  25  *******
 918  001610  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 919  001612  000000                          HALT                    ;PSW NOT 105
 920                                                                  ;  OR SEQUENCE ERROR
 921
 922                                  ;*****************************************************************************
 923                                  ;TEST 23        TEST IF PSW (EXCEPT T-BIT) WILL HOLD ALL ONES
 924                                  ;*****************************************************************************
 925  001614  005212                  TST23:  INC     (R2)            ;UPDATE TEST NUMBER
 926  001616  022712  000023                  CMP     #23,(R2)        ;SEQUENCE ERROR?
 927  001622  001007                          BNE     TST24-10        ;BR TO ERROR HALT ON SEQ ERROR
 928  001624  012737  000357  177776          MOV     #357,@#PS       ;MOVE ONES TO PSW
 929  001632  023727  177776  000357          CMP     @#PS,#357       ;SUCCESSFUL
 930  001640  001404                          BEQ     TST24
 931                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 932                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
 933                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
 934                                                                  ;           WHICH FOLLOWS W/ 771           <====
 935  001642  012742  000026                  MOV     #26,-(R2)       ;MOVE TO MAILBOX #  *******  26  *******
 936  001646  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 937  001650  000000                          HALT                    ;PSW NOT 357
 938                                                                  ;  OR SEQUENCE ERROR
```

```
                                        ;*************************************************************************
                                        .SBTTL  B-REGISTER TEST
                                        ;
                                        ;       THE B-REGISTER SHIFTING LOGIC TESTS ARE USED TO TEST THAT THE
                                        ;B-REGISTER CAN HOLD VARIOUS DATA PATTERNS AND THAT THE ASSOCIATED
                                        ;LOGIC SUPPORTS THE SHIFTING FUNCTIONS WITHIN THE B-REGISTER AND C-BIT.
                                        ;A ONE IS SHIFTED THROUGH EVERY BIT IN THE B-REGISTER AND C-BIT IN
                                        ;BOTH DIRECTIONS.
                                        ;       THE B-REGISTER ITSELF IS TESTED IN ITS ABILITY AS A BUFFER AND AS
                                        ;A SHIFT REGISTER.  DATA IS ALSO PASSED THROUGH THE DATA PATH AND ALU.
                                        ;       IF THESE TESTS FAIL, EXAMINE THE TARGET LOCATION (LOC. 0) TO SEE
                                        ;WHICH BITS OF THE B-REGISTER MAY BE FAILING.  IF THIS PROVIDES
                                        ;INCONCLUSIVE DATA TRY TO CHECK THE MODE 3 IR DECODE BY RUNNING JUST
                                        ;THE MICROCODE AND IR DECODE TESTS FOR THE PARTICULAR INSTRUCTIONS.
                                        ;
                                        ;*************************************************************************
                                        ;TEST 24        SHIFT BIT 0 TO BIT 1
                                        ;*************************************************************************
001652  005212                          TST24:  INC     (R2)            ;UPDATE TEST NUMBER
001654  022712  000024                          CMP     #24,(R2)        ;SEQUENCE ERROR?
001660  001012                                  BNE     TST25-10        ;BR TO ERROR HALT ON SEQ ERROR
001662  000241                                  CLC                     ;CLEAR CARRY BIT
001664  012737  000001  000000                  MOV     #1,@#0          ;LOAD A 1
001672  006137  000000                          ROL     @#0             ;SHIFT LEFT
001676  022737  000002  000000                  CMP     #2,@#0          ;SUCCESSFUL
001704  001404                                  BEQ     TST25
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                        ;           WHICH FOLLOWS W/ 766            <====
001706  012742  000027                          MOV     #27,-(R2)       ;MOVE TO MAILBOX #  *******  27  *******
001712  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001714  000000                                  HALT                    ;BIT 1 NOT SET
                                                                        ;  OR SEQUENCE ERROR

                                        ;*************************************************************************
                                        ;TEST 25        SHIFT CARRY INTO BIT 0
                                        ;*************************************************************************
001716  005212                          TST25:  INC     (R2)            ;UPDATE TEST NUMBER
001720  022712  000025                          CMP     #25,(R2)        ;SEQUENCE ERROR?
001724  001017                                  BNE     TST26-10        ;BR TO ERROR HALT ON SEQ ERROR
001726  012737  000000  000000                  MOV     #0,@#0          ;CLEAR LOCATION
001734  000261                                  SEC                     ;SET CARRY
001736  006137  000000                          ROL     @#0             ;ROTATE CARRY BIT TO BIT 0
001742  103014                                  BCC     TST26
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                        ;           WHICH FOLLOWS W/ 771            <====
001744  012742  000030                          MOV     #30,-(R2)       ;MOVE TO MAILBOX #  *******  30  *******
001750  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001752  000000                                  HALT                    ;CARRY CLEAR
                                                                        ;  OR SEQUENCE ERROR
001754  022737  000001  000000                  CMP     #1,@#0          ;BIT 0 SET
001762  001404                                  BEQ     TST26
```

```
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                        ;           CONDITIONAL BRANCH INST. AND      <====
                                                        ;           REPLACE THE MOVE INSTRUCTION      <====
                                                        ;           WHICH FOLLOWS W/ 761              <====
001764  012742  000031                  MOV     #31,-(R2)       ;MOVE TO MAILBOX #  *******  31  *******
001770  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
001772  000000                          HALT                    ;BIT 0 NOT SET
                                                                ;  OR SEQUENCE ERROR

                                ;*************************************************************************
                                ;TEST 26        LEFT SHIFT FROM BIT 0 TO C-BIT
                                ;*************************************************************************
001774  005212                  TST26:  INC     (R2)            ;UPDATE TEST NUMBER
001776  022712  000026                  CMP     #26,(R2)        ;SEQUENCE ERROR?
002002  001010                          BNE     TST27-10        ;BR TO ERROR HALT ON SEQ ERROR
002004  012737  000001  000000          MOV     #1,@#0          ;SET BIT 0
002012  000241                          CLC                     ;CLEAR C-BIT
002014  006137  000000          SHL:    ROL     @#0             ;SHIFT LEFT ONE POSITION
002020  103375                          BCC     SHL             ;BRANCH IF C-BIT NOT SET
002022  001404                          BEQ     TST27
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                        ;           CONDITIONAL BRANCH INST. AND      <====
                                                        ;           REPLACE THE MOVE INSTRUCTION      <====
                                                        ;           WHICH FOLLOWS W/ 770              <====
002024  012742  000032                  MOV     #32,-(R2)       ;MOVE TO MAILBOX #  *******  32  *******
002030  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002032  000000                          HALT                    ;LEFT SHIFTING LOGIC FAILED
                                                                ;  OR SEQUENCE ERROR

                                ;*************************************************************************
                                ;TEST 27        SHIFT BIT 15 TO BIT 14
                                ;*************************************************************************
002034  005212                  TST27:  INC     (R2)            ;UPDATE TEST NUMBER
002036  022712  000027                  CMP     #27,(R2)        ;SEQUENCE ERROR?
002042  001012                          BNE     TST30-10        ;BR TO ERROR HALT ON SEQ ERROR
002044  012737  100000  000000          MOV     #100000,@#0     ;SET BIT 15
002052  000241                          CLC                     ;CLEAR CARRY
002054  006037  000000                  ROR     @#0             ;SHIFT BIT 15 TO BIT 14
002060  022737  040000  000000          CMP     #40000,@#0      ;SUCCESSFUL
002066  001404                          BEQ     TST30
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                        ;           CONDITIONAL BRANCH INST. AND      <====
                                                        ;           REPLACE THE MOVE INSTRUCTION      <====
                                                        ;           WHICH FOLLOWS W/ 766              <====
002070  012742  000033                  MOV     #33,-(R2)       ;MOVE TO MAILBOX #  *******  33  *******
002074  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002076  000000                          HALT                    ;BIT 14 NOT SET
                                                                ;  OR SEQUENCE ERROR

                                ;*************************************************************************
                                ;TEST 30        RIGHT SHIFT FROM BIT 15 TO C-BIT
                                ;*************************************************************************
002100  005212                  TST30:  INC     (R2)            ;UPDATE TEST NUMBER
002102  022712  000030                  CMP     #30,(R2)        ;SEQUENCE ERROR?
002106  001010                          BNE     TST31-10        ;BR TO ERROR HALT ON SEQ ERROR
002110  012737  100000  000000          MOV     #100000,@#0             ;SET BIT 15
```

```
1051  002116  000241                    CLC             ;CLEAR C-BIT
1052  002120  006037  000000  SHP:      ROR     @#0     ;ROTATE RIGHT ONE POSITION
1053  002124  103375                    BCC     SHR     ;BRANCH IF C-BIT CLEAR
1054  002126  001404                    BEQ     TST31
1055                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1056                                                    ;           CONDITIONAL BRANCH INST. AND   <====
1057                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
1058                                                    ;           WHICH FOLLOWS W/ 770           <====
1059  002130  012742  000034                    MOV     #34,-(R2)       ;MOVE TO MAILBOX #  *******  34  *******
1060  002134  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1061  002136  000000                    HALT                    ;RIGHT SHIFT LOGIC FAILED
1062                                                            ;  OR SEQUENCE ERROR
```

```
                           .SBTTL  CONDITION CODE TEST

                           ;*****************************************************************
                           ;
                           ;       THIS TEST CHECKS THE CONDITIONAL BRANCHES INVOLVING THE Z-BIT.
                           ;THE Z-BIT IS SET WITH ALL OTHER CC BITS ZERO AND BOTH CONDITIONS
                           ;BEQ AND BNE ARE TESTED FOR PROPER EXECUTION.  THEN THE Z-BIT IS
                           ;SET WITH ALL OTHER CC BITS CLEAR AND BOTH CONDITIONS ARE TESTED
                           ;AGAIN FOR PROPER OPERATION.
                           ;       THIS TEST CHECKS THE OPERATION OF THE SET AND CLEAR CONDITION
                           ;CODE INSTRUCTIONS AND CHECKS THE CIRCUITRY EXTERNAL TO THE CONDITIONAL
                           ;BRANCH ROM.  THE BRANCH MICROCODE FOR ALTERING THE PC AND FOR
                           ;LEAVING THE PC UNALTERED IS TESTED.  ONLY THOSE ROM ADDRESSES SPECIFICALLY
                           ;USED IN THE TEST ARE VERIFIED HERE.
                           ;
                           ;*****************************************************************
                           ;TEST 31         TEST BRANCHES AROUND Z-BIT
                           ;*****************************************************************
002140  005212             TST31:  INC     (R2)            ;UPDATE TEST NUMBER
002142  022712  000031             CMP     #31,(R2)        ;SEQUENCE ERROR?
002146  001014                     BNE     TST32-10        ;BR TO ERROR HALT ON SEQ ERROR
                                   ;FIRST WITH Z-BIT ON
002150  000257                     CCC                     ;CC=0100: JUST Z-BIT
002152  000264                     SEZ
002154  001001                     BNE     BRZ1            ;CHECK OPPOSITE CONDITION
002156  001404                     BEQ     BRZ2
                                                           ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                           ;           CONDITIONAL BRANCH INST. AND    <====
                                                           ;           REPLACE THE MOVE INSTRUCTION    <====
                                                           ;           WHICH FOLLOWS W/ 774            <====
002160                     BRZ1:
002160  012742  000035             MOV     #35,-(R2)       ;MOVE TO MAILBOX #  *******  35  *******
002164  005242                     INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002166  000000                     HALT                    ;IMPROPER BR W/ Z=1
                                   ;CHECK WITH Z-BIT OFF
002170  000277             BRZ2:   SCC                     ;CC=1011: ALL BUT Z-BIT
002172  000244                     CLZ
002174  001401                     BEQ     BRZ3
002176  001004                     BNE     TST32
                                                           ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                           ;           CONDITIONAL BRANCH INST. AND    <====
                                                           ;           REPLACE THE MOVE INSTRUCTION    <====
                                                           ;           WHICH FOLLOWS W/ 764            <====
002200                     BRZ3:
002200  012742  000036             MOV     #36,-(R2)       ;MOVE TO MAILBOX #  *******  36  *******
002204  005242                     INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002206  000000                     HALT                    ;IMPROPER BR W/ Z=0
                                                           ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST CHECKS THE CONDITIONAL BRANCHES INVOLVING THE N-BIT.
                                ;THE N-BIT IS SET WITH ALL OTHER CC BITS ZERO AND BOTH CONDITIONS
                                ;BMI AND BPL ARE TESTED FOR PROPER EXECUTION.  THEN THE N-BIT IS
                                ;SET WITH ALL OTHER CC BITS CLEAR AND BOTH CONDITIONS ARE TESTED
                                ;AGAIN FOR PROPER OPERATION.
                                ;       THIS TEST CHECKS THE OPERATION OF THE SET AND CLEAR CONDITION
                                ;CODE INSTRUCTIONS AND CHECKS THE CIRCUITRY EXTERNAL TO THE CONDITIONAL
                                ;BRANCH ROM.  THE BRANCH MICROCODE FOR ALTERING THE PC AND FOR
                                ;LEAVING THE PC UNALTERED IS TESTED.  ONLY THOSE ROM ADDRESSES SPECIFICALLY
                                ;USED IN THE TEST ARE VERIFIED HERE.
                                ;
                                ;*****************************************************************************
                                ;TEST 32         TEST BRANCHES AROUND N-BIT
                                ;*****************************************************************************
002210  005212                TST32:  INC     (R2)            ;UPDATE TEST NUMBER
002212  022712  000032                CMP     #32,(R2)        ;SEQUENCE ERROR?
002216  001014                        BNE     TST33-10        ;BR TO ERROR HALT ON SEQ ERROR
                                        ;FIRST WITH N-BIT ON
002220  000257                        CCC                     ;CC=1000: JUST N-BIT
002222  000270                        SEN
002224  100001                        BPL     BRN1            ;CHECK OPPOSITE CONDITION
002226  100404                        BMI     BRN2
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 774           <====
002230                          BRN1:
002230  012742  000037                MOV     #37,-(R2)       ;MOVE TO MAILBOX #  *******  37  *******
002234  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002236  000000                        HALT                    ;IMPROPER BR W/ N=1
                                        ;CHECK WITH N-BIT OFF
002240  000277                BRN2:   SCC                     ;CC=0111
002242  000250                        CLN
002244  100401                        BMI     BRN3            ;CHECK OPPOSITE CONDITION
002246  100004                        BPL     TST33
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 764           <====
002250                          BRN3:
002250  012742  000040                MOV     #40,-(R2)       ;MOVE TO MAILBOX #  *******  40  *******
002254  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002256  000000                        HALT                    ;IMPROPER BR W/ N=0
                                                            ;  OR SEQUENCE ERROR
```

```
1158
1159                    ;*************************************;*************************************************
1160                    ;
1161                    ;       THIS TEST CHECKS THE CONDITIONAL BRANCHES INVOLVING THE V-BIT.
1162                    ;THE V-BIT IS SET WITH ALL OTHER CC BITS ZERO AND BOTH CONDITIONS
1163                    ;BVS AND BVC ARE TESTED FOR PROPER EXECUTION.  THEN THE V-BIT IS
1164                    ;SET WITH ALL OTHER CC BITS CLEAR AND BOTH CONDITIONS ARE TESTED
1165                    ;AGAIN FOR PROPER OPERATION.
1166                    ;       THIS TEST CHECKS THE OPERATION OF THE SET AND CLEAR CONDITION
1167                    ;CODE INSTRUCTIONS AND CHECKS THE CIRCUITRY EXTERNAL TO THE CONDITIONAL
1168                    ;BRANCH ROM.  THE BRANCH MICROCODE FOR ALTERING THE PC AND FOR
1169                    ;LEAVING THE PC UNALTERED IS TESTED.  ONLY THOSE ROM ADDRESSES SPECIFICALLY
1170                    ;USED IN THE TEST ARE VERIFIED HERE.
1171                    ;
1172                    ;*****************************************************************************
1173                    ;TEST 33        TEST BRANCHES AROUND V-BIT
1174                    ;*****************************************************************************
1175  002260  005212    TST33:  INC     (R2)            ;UPDATE TEST NUMBER
1176  002262  022712    000033          CMP     #33,(R2)        ;SEQUENCE ERROR?
1177  002266  001014                    BNE     TST34-10        ;BR TO ERROR HALT ON SEQ ERROR
1178                                    ;FIRST WITH V-BIT ON
1179  002270  000257                    CCC                     ;CC=0010: JUST V-BIT
1180  002272  000262                    SEV
1181  002274  102001                    BVC     BRV1            ;CHECK OPPOSITE CONDITION
1182  002276  102404                    BVS     BRV2
1183                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1184                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1185                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1186                                                            ;           WHICH FOLLOWS W/ 774           <====
1187  002300            BRV1:
1188  002300  012742    000041          MOV     #41,-(R2)       ;MOVE TO MAILBOX #  *******  41  *******
1189  002304  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1190  002306  000000                    HALT                    ;IMPROPER BR W/ V=1
1191                                    ;CHECK WITH V-BIT OFF
1192  002310  000277    BRV2:           SCC                     ;CC=1101: ALL BVT V-BIT
1193  002312  000242                    CLV
1194  002314  102401                    BVS     BRV3            ;CHECK OPPOSITE CONDITION
1195  002316  102004                    BVC     TST34
1196                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1197                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1198                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1199                                                            ;           WHICH FOLLOWS W/ 764           <====
1200  002320            BRV3:
1201  002320  012742    000042          MOV     #42,-(R2)       ;MOVE TO MAILBOX #  *******  42  *******
1202  002324  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1203  002326  000000                    HALT                    ;IMPROPER BR W/ V=0
1204                                                            ;  OR SEQUENCE ERROR
```

```
1205
1206                            ;*****************************************************************************
1207                            ;
1208                            ;       THIS TEST CHECKS THE CONDITIONAL BRANCHES INVOLVING THE C-BIT.
1209                            ;THE C-BIT IS SET WITH ALL OTHER CC BITS ZERO AND BOTH CONDITIONS
1210                            ;BCS AND BCC ARE TESTED FOR PROPER EXECUTION.  THEN THE C-BIT IS
1211                            ;SET WITH ALL OTHER CC BITS CLEAR AND BOTH CONDITIONS ARE TESTED
1212                            ;AGAIN FOR PROPER OPERATION.
1213                            ;       THIS TEST CHECKS THE OPERATION OF THE SET AND CLEAR CONDITION
1214                            ;CODE INSTRUCTIONS AND CHECKS THE CIRCUITRY EXTERNAL TO THE CONDITIONAL
1215                            ;BRANCH ROM.  THE BRANCH MICROCODE FOR ALTERING THE PC AND FOR
1216                            ;LEAVING THE PC UNALTERED IS TESTED.  ONLY THOSE ROM ADDRESSES SPECIFICALLY
1217                            ;USED IN THE TEST ARE VERIFIED HERE.
1218                            ;
1219                            ;*****************************************************************************
1220                            ;TEST 34        TEST BRANCHES AROUND C-BIT
1221                            ;*****************************************************************************
1222  002330  005212            TST34:  INC     (R2)            ;UPDATE TEST NUMBER
1223  002332  022712  000034            CMP     #34,(R2)        ;SEQUENCE ERROR?
1224  002336  001014                    BNE     TST35-10        ;BR TO ERROR HALT ON SEQ ERROR
1225                                    ;FIRST WITH C-BIT ON
1226  002340  000257                    CCC                     ;CC=0001: JUST C-BIT
1227  002342  000261                    SEC
1228  002344  103001                    BCC     BRC1            ;CHECK OPPOSITE CONDITION
1229  002346  103404                    BCS     BRC2
1230                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1231                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1232                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1233                                                            ;           WHICH FOLLOWS W/ 774           <====
1234  002350                    BRC1:
1235  002350  012742  000043           MOV     #43,-(R2)       ;MOVE TO MAILBOX #  *******  43  *******
1236  002354  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1237  002356  000000                    HALT                    ;IMPROPER BR W/ C=1
1238                                    ;CHECK WITH V-BIT OFF
1239  002360  000277            BRC2:   SCC                     ;CC=1110
1240  002362  000242                    CLV
1241  002364  102401                    BVS     BRC3            ;CHECK OPPOSITE CONDITION
1242  002366  100404                    BMI     TST35
1243                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1244                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1245                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1246                                                            ;           WHICH FOLLOWS W/ 764           <====
1247  002370                    BRC3:
1248  002370  012742  000044           MOV     #44,-(R2)       ;MOVE TO MAILBOX #  *******  44  *******
1249  002374  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1250  002376  000000                    HALT                    ;IMPROPER BR W/ C=0
1251                                                            ;  OR SEQUENCE ERROR
```

```
1252
1253                            ;*****************************************************************************
1254                            .SBTTL  MICROCODE TESTS
1255                            ;
1256                            ;       THE MICROCODE TESTS ARE USED TO VERIFY THE MICROPROGRAMM
1257                            ;FLOW.  THE GOAL OF THESE TESTS IS TO EXERCISE EVERY POSSIBLE
1258                            ;BRANCH IN THE MICROPROGRAM FLOW.
1259                            ;       THE TEST EXERCISES EVERY BRANCH IN THE MICROCODE BY
1260                            ;TESTING AT LEAST ONE INSTRUCTION FROM EVERY CLASS OF INSTRUCTION IN
1261                            ;ALL POSSIBLE MODES.  FOR EXAMPLE, TO TEST THE SINGLE OPERAND INSTRUCTIONS,
1262                            ;AT LEAST ONE SINGLE OPERAND INSTRUCTION IS VERIFIED IN ALL UNIQUE
1263                            ;ADDRESSING MODES.  BYTE MODES ARE ALSO TESTED.  AS EACH NEW
1264                            ;MODE IS INTRODUCED THE SAME INSTRUCTION IS TRIED AND TESTED IN
1265                            ;A SMALL LOOP CONVENIENT FOR SCOPING.  THE TEST IS SET UP USING
1266                            ;ONLY INSTRUCTIONS AND ADDRESSING MODES WHICH HAVE BEEN PREVIOUSLY
1267                            ;VERIFIED.
1268                            ;       IF THESE TESTS FAIL, CHECK THE RESULTS FOR A CLUE TO THE
1269                            ;FAULT.
1270                            ;
1271                            ;*****************************************************************************
1272
1273
1274
1275                            ;*****************************************************************************
1276                            ;
1277                            ;       THE CLR INSTRUCTION IS USED TO INTRODUCE EACH ADDRESSING
1278                            ;MODE WITH THE SINGLE OPERAND INSTRUCTION.  FOLLOWING THE SEQUENCE CHECK,
1279                            ;THE CLR INSTRUCTION IS EXECUTED AND A BRANCH TEST IS EXECUTED WHICH
1280                            ;CHECKS THAT THE Z-BIT WAS PROPERLY SET.  THIS SMALL TEST IS SELF-SUFFICIENT
1281                            ;AND CAN BE SCOPED TO TROUBLE SHOOT ALL OF THE IR DECODE LOGIC AND
1282                            ;MICROCODE FOR SOP INSTRUCTIONS WITH MODE 0.  FOLLOWING THIS TEST
1283                            ;SEVERAL OTHER SOP INSTRUCTIONS ARE INTRODUCED WITH MODE 0.  THESE
1284                            ;INSTRUCTIONS MAINPULATE DATA AND SERVE TO CHECK THE DATA RESULTS
1285                            ;OF THE SOP INSTRUCTIONS IN THIS TEST.  THE DATA IN THIS TEST IS
1286                            ;OPERATED ON BY EACH INSTRUCTION WITHOUT REINITIALIZING.
1287                            ;
1288                            ;*****************************************************************************
1289                            ;TEST 35        TEST MODE 0 USING SOP INST.
1290                            ;*****************************************************************************
1291  002400  005212            TST35:  INC     (R2)            ;UPDATE TEST NUMBER
1292  002402  022712  000035            CMP     #35,(R2)        ;SEQUENCE ERROR?
1293  002406  001017                    BNE     TST36-10        ;BR TO ERROR HALT ON SEQ ERROR
1294  002410  005000                    CLR     R0              ;TRY THE CLEAR INST.
1295  002412  001404                    BEQ     SOPOA
1296                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1297                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1298                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1299                                                            ;           WHICH FOLLOWS W/ 776           <====
1300  002414  012742  000045            MOV     #45,-(R2)       ;MOVE TO MAILBOX #  *******  45  *******
1301  002420  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1302  002422  000000                    HALT                    ;CLR DID NOT SET Z-BIT
1303  002424  005200            SOPOA:  INC     R0              ;TRY THE INCREMENT INST.
1304  002426  005400                    NEG     R0              ;TRY THE NEGATE INST.
1305  002430  100404 ↑                  BMI     SOPOB
1306                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1307                                                            ;           CONDITIONAL BRANCH INST. AND   <====
```

```
1308                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1309                                                            ;           WHICH FOLLOWS W/ 767           <====
1310  002432  012742  000046                MOV     #46,-(R2)  ;MOVE TO MAILBOX #  *******  46  *******
1311  002436  005242                          INC     -(R2)      ;SET MSGTYP TO FATAL ERROR
1312  002440  000000                          HALT               ;NEGATE DID NOT SET N-BIT
1313  002442  005100                SOP0B:    COM     R0         ;TRY COMPLEMENT INST.
1314  002444  001404                          BEQ     TST36
1315                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1316                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1317                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1318                                                            ;           WHICH FOLLOWS W/ 761           <====
1319  002446  012742  000047                MOV     #47,-(R2)  ;MOVE TO MAILBOX #  *******  47  *******
1320  002452  005242                          INC     -(R2)      ;SET MSGTYP TO FATAL ERROR
1321  002454  000000                          HALT               ;CUMMULATIVE RESULT OF CLR,INC,NEG AND COM INSTS. FAILED
1322                                                            ;  OR SEQUENCE ERROR
1323
1324
1325                                ;*************************************************************************
1326                                ;
1327                                ;       THIS TEST INTRODUCES THE REMAINING SOP INSTRUCIONS AND TESTS
1328                                ;THEM IN MODE 0.  THE PURPOSE IS TO PROVIDE A BASELINE OF
1329                                ;INSTRUCTIONS FOR USE IN THE SUBSEQUENT TESTS.  SINCE THE MICROCODE FOR
1330                                ;THESE INSTRUCTIONS IS IDENTICAL TO THAT ALREADY TESTED, ANY TROUBLE
1331                                ;SHOOTING EFFORTS SHOULD BE AIMED AT THE ACTUAL IR DECODE AND ALU
1332                                ;FUNCTIONING.
1333                                ;
1334                                ;*************************************************************************
1335                                ;TEST 36        TEST REMAINDER OF SOP INSTS IN MODE 0
1336                                ;*************************************************************************
1337  002456  005212                TST36:    INC     (R2)       ;UPDATE TEST NUMBER
1338  002460  022712  000036                  CMP     #36,(R2)   ;SEQUENCE ERROR?
1339  002464  001020                          BNE     TST37-10   ;BR TO ERROR HALT ON SEQ ERROR
1340  002466  005000                          CLR     R0         ;INITIALIZE
1341  002470  005300                          DEC     R0         ;TRY DECREMENT INST.
1342  002472  100404                          BMI     SOP0C
1343                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1344                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1345                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1346                                                            ;           WHICH FOLLOWS W/ 775           <====
1347  002474  012742  000050                MOV     #50,-(R2)  ;MOVE TO MAILBOX #  *******  50  *******
1348  002500  005242                          INC     -(R2)      ;SET MSGTYP TO FATAL ERROR
1349  002502  000000                          HALT               ;N-BIT NOT SET ON DEC
1350  002504  000261                SOP0C:    SEC                ;INITIALIZE CARRY
1351  002506  005500                          ADC     R0         ;TRY ADD CARRY INST
1352  002510  001006                          BNE     SOP0D
1353  002512  000261                          SEC                ;INITIALIZE CARRY
1354  002514  005600                          SBC     R0         ;TRY SUBTRACT-CARRY INST
1355  002516  100003                          BPL     SOP0D
1356  002520  005400                          NEG     R0
1357  002522  005300                          DEC     R0
1358  002524  001404                          BEQ     TST37
1359                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1360                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1361                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1362                                                            ;           WHICH FOLLOWS W/ 760           <====
1363  002526                        SOP0D:
```

```
 1364  002526  012742  000051                  MOV     #51,-(R2)       ;MOVE TO MAILBOX #  ******  51  *******
 1365  002532  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 1366  002534  000000                          HALT                    ; CUMMULATIVE RESULT OF ADC,SBC,NEG AND DEC INSTS. FAILE
 1367                                                                  ;  OR SEQUENCE ERROR
```

```
1368
1369                                    ;*****************************************************************************
1370                                    ;
1371                                    ;       THIS TEST INTRODUCES THE BYTE CONTROL LOGIC OF THE PROCESSOR.
1372                                    ;THE MODE 0 BYTE MICROCODE IS TESTED.  THE METHOD AND SEQUENCE
1373                                    ;OF TESTING IS THE SAME AS THAT USED IN THE SOP MODE 0 TESTS.
1374                                    ;
1375                                    ;*****************************************************************************
1376                                    ;*****************************************************************************
1377                                    ;TEST 37        TEST MODE 0 EVEN BYTE USING SOP INST
1378                                    ;*****************************************************************************
1379 002536  005212                    TST37:  INC     (R2)            ;UPDATE TEST NUMBER
1380 002540  022712  000037            CMP     #37,(R2)        ;SEQUENCE ERROR?
1381 002544  001012                    BNE     TST40-10        ;BR TO ERROR HALT ON SEQ ERROR
1382 002546  105000                    CLRB    R0              ;TRY CLEARING EVEN BYTE OF REGISTER
1383 002550  001404                    BEQ     SOPB0A
1384                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1385                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1386                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1387                                                            ;           WHICH FOLLOWS W/ 776           <====
1388 002552  012742  000052            MOV     #52,-(R2)       ;MOVE TO MAILBOX #  *******  52  *******
1389 002556  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1390 002560  000000                    HALT                    ;CLRB DID NOT SET Z-BIT
1391 002562  105100            SOPB0A: COMB    R0              ;TRY SETTING EVEN BYTE OF REGISTER
1392 002564  100002                    BPL     SOPB0B
1393 002566  105200                    INCB    R0              ;TRY INCREMENTING EVEN BYTE OF REGISTER))
1394 002570  001404                    BEQ     TST40
1395                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1396                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1397                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1398 002572                            SOPB0B:                 ;           WHICH FOLLOWS W/ 766           <====
1399 002572  012742  000053            MOV     #53,-(R2)       ;MOVE TO MAILBOX #  *******  53  *******
1400 002576  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1401 002600  000000                    HALT                    ;TEST CUMMULATIVE RESULT OF ABOVE BYTE INST.
1402                                                            ;  OR SEQUENCE ERROR
```

```
                        ;*****************************************************************************
                        ;
                        ;       THIS TEST USES THE CLR INSTRUCTION TO INTRODUCE AND TEST
                        ;SINGLE OPERAND MODE 1 INSTRUCTIONS.  AGAIN, THE CLR INSTRUCTION
                        ;IS USED TO INTRODUCE THE MICROCODE AND TO TEST THAT THE PROPER
                        ;CONDITION CODES ARE SET.  OTHER SOP INSTRUCTIONS ARE USED TO MANIPULATE
                        ;COMMON DATA TO VERIFY THAT THE CORRECT DATA IS PRODUCED.
                        ;
                        ;*****************************************************************************
                        ;TEST 40        TEST MODE 1 USING SOP INST.
                        ;*****************************************************************************
002602  005212         TST40:  INC     (R2)            ;UPDATE TEST NUMBER
002604  022712  000040         CMP     #40,(R2)        ;SEQUENCE ERROR?
002610  001014                 BNE     TST41-10        ;BR TO ERROR HALT ON SEQ ERROR
002612  005000                 CLR     R0              ;INITIALIZE R0
002614  005010                 CLR     (R0)            ;TRY CLEAR INST W/MODE 1
002616  001404                 BEQ     SOP1A
                                                       ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                       ;           CONDITIONAL BRANCH INST. AND   <====
                                                       ;           REPLACE THE MOVE INSTRUCTION   <====
                                                       ;           WHICH FOLLOWS W/ 775           <====
002620  012742  000054                 MOV     #54,-(R2)       ;MOVE TO MAILBOX #  *******  54  *******
002624  005242                 INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002626  000000                 HALT                    ;CLR DID NOT SET Z-BIT
002630  005310         SOP1A:  DEC     (R0)            ;TRY DECREMENT INST W/MODE 1
002632  100003                 BPL     SOP1B
002634  000261                 SEC                     ;INITIALIZE CARRY
002636  005510                 ADC     (R0)            ;TRY ADD-CARRY W/MODE 1
002640  001404                 BEQ     TST41
                                                       ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                       ;           CONDITIONAL BRANCH INST. AND   <====
                                                       ;           REPLACE THE MOVE INSTRUCTION   <====
                                                       ;           WHICH FOLLOWS W/ 764           <====
002642         SOP1B:
002642  012742  000055                 MOV     #55,-(R2)       ;MOVE TO MAILBOX #  *******  55  *******
002646  005242                 INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002650  000000                 HALT                    ;TEST CUMMULATIVE RESULT OF ABOVE INST
                                                       ;  OR SEQUENCE ERROR
```

```
                                     ;*****************************************************************
                                     ;
                                     ;       THIS TEST VERIFIES THE BYTE INSTRUCTION MICROCODE FOR MODE 1
                                     ;SINGLE OPERAND INSTRUCTIONS.
                                     ;       THIS IS THE FIRST PLACE THE SIGN EXTEND LOGIC IS EXERCISED
                                     ;AND VERIFIED.
                                     ;
                                     ;*****************************************************************
                                     ;TEST 41        TEST MODE 1 EVEN BYTE USING SOP INST
                                     ;*****************************************************************
002652  005212                      TST41:  INC     (R2)            ;UPDATE TEST NUMBER
002654  022712  000041                      CMP     #41,(R2)        ;SEQUENCE ERROR?
002660  001017                              BNE     TST42-10        ;BR TO ERROR HALT ON SEQ ERROR
002662  005000                              CLR     R0              ;INITIALIZE R0
002664  005010                              CLR     (R0)            ;INITIALIZE LOC. 0
002666  005110                              COM     (R0)
002670  105010                              CLRB    (R0)            ;TRY TO CLEAR BYTE 0
002672  001404                              BEQ     SOPB1A
                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                    ;           CONDITIONAL BRANCH INST. AND    <====
                                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                    ;           WHICH FOLLOWS W/ 773            <====
002674  012742  000056                      MOV     #56,-(R2)       ;MOVE TO MAILBOX #  *******  56  *******
002700  005242                              INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002702  000000                              HALT                    ;CLRB DID NOT SET Z-BIT
002704  005210                      SOPB1A: INC     (R0)            ;INCREMENT TO TEST WORD
002706  100004                              BPL     SOPB1B
002710  105410                              NEGB    (R0)            ;NEGATE: ODD BYTE=377
002712  100002                              BPL     SOPB1B
002714  105210                              INCB    (R0)            ;INCREMENT ODD BYTE=0
002716  001404                              BEQ     TST42
                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                    ;           CONDITIONAL BRANCH INST. AND    <====
                                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                    ;           WHICH FOLLOWS W/ 761            <====
002720                              SOPB1B:
002720  012742  000057                      MOV     #57,-(R2)       ;MOVE TO MAILBOX #  *******  57  *******
002724  005242                              INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002726  000000                              HALT                    ;CHECK CUMMULATIVE RESULT OF ABOVE INST
                                                                    ;  OR SEQUENCE ERROR
```

```
                                ;*************************************************************************
                                ;
                                ;       THIS TEST VERIFIES THAT SINGLE OPERAND BYTE INSTRUCTIONS WILL
                                ;FUNCTION CORRECTLY FOR ODD BYTES.
                                ;       THIS IS THE FIRST TIME THAT ADDRESS LINE 0 HAS BEEN
                                ;EXERCISED.  CHECKS ARE MADE THAT THE PROPER BYTE IS MODIFIED AND
                                ;THE CONDITION CODES ARE CHECKED.  IT IS ALSO VERIFIED THAT THE UNADDRESSED
                                ;BYTE IS NOT ALTERED BY THE INSTRUCTION.
                                ;
                                ;*************************************************************************
                                ;TEST 42        TEST MODE 1 ODD BYTE USING SOP INST
                                ;*************************************************************************
002730  005212                  TST42:  INC     (R2)            ;UPDATE TEST NUMBER
002732  022712  000042                  CMP     #42,(R2)        ;SEQUENCE ERROR?
002736  001021                          BNE     TST43-10        ;BR TO ERROR HALT ON SEQ ERROR
002740  005000                          CLR     R0              ;INITIALIZE R0
002742  005010                          CLR     (R0)            ;INITIALIZE LOC. 0
002744  005110                          COM     (R0)
002746  005200                          INC     R0              ;R0=ODD BYTE
002750  105010                          CLRB    (R0)            ;TRY TO CLEAR BYTE 1
002752  001404                          BEQ     SOPB1C
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                ;           CONDITIONAL BRANCH INST. AND    <====
                                                                ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                ;           WHICH FOLLOWS W/ 772            <====
002754  012742  000060                  MOV     #60,-(R2)       ;MOVE TO MAILBOX #  *******  60  *******
002760  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
002762  000000                          HALT                    ;CLRB DID NOT SET Z-BIT
002764  005300                  SOPB1C: DEC     R0              ;R0=WORD ADDR.
002766  005210                          INC     (R0)            ;INCREMENT TO TEST WORD
002770  005200                          INC     R0              ;R0=ODD BYTE
002772  105410                          NEGB    (R0)            ;TRY TO NEGATE BYTE 1
002774  100002                          BPL     SOPB1D
002776  105210                          INCB    (R0)            ;TRY TO INCREMENT BYTE 1
003000  001404                          BEQ     TST43
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                ;           CONDITIONAL BRANCH INST. AND    <====
                                                                ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                ;           WHICH FOLLOWS W/ 757            <====
003002                          SOPB1D:
003002  012742  000061                  MOV     #61,-(R2)       ;MOVE TO MAILBOX #  *******  61  *******
003006  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003010  000000                          HALT                    ;TEST CUMMULATIVE RESULT OF ABOVE INST.
                                                                ;  OR SEQUENCE ERROR
```

```
                                        ;*********************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES MODE 2 SINGLE-OPERAND INSTRUCTIONS.  PREVIOUSLY
                                        ;TESTED INSTRUCTIONS ARE USED TO SET A POINTER IN R0 TO LOC. 400.
                                        ;LOC. 400 IS INITIALIZED TO -1 BEFORE A CLR MODE 2 IS EXECUTED.
                                        ;       THEN R0 IS DECREMENTED BY TWO TO AGAIN POINT TO 400 BEFORE EACH
                                        ;OF SEVERAL MODE 2 INSTRUCTIONS ARE USED TO VERIFY THE DATA RESULTS OF
                                        ;THE TEST.  THIS PROCEDURE ALSO VERIFIES THE PROPER INCREMENTING OF THE
                                        ;REGISTER.
                                        ;
                                        ;*********************************************************************
                                        ;TEST 43        TEST MODE 2 USING SOP INST.
                                        ;*********************************************************************
003012  005212                          TST43:  INC     (R2)            ;UPDATE TEST NUMBER
003014  022712  000043                          CMP     #43,(R2)        ;SEQUENCE ERROR?
003020  001023                                  BNE     TST44-10        ;BR TO ERROR HALT ON SEQ ERROR
003022  005000                                  CLR     R0              ;SET R0=400
003024  105100                                  COMB    R0
003026  005200                                  INC     R0
003030  005010                                  CLR     (R0)            ;CLEAR 400
003032  005110                                  COM     (R0)            ;INITIALIZE: 400=-1
003034  005020                                  CLR     (R0)+           ;TRY CLEARING WITH MODE 2
003036  001404                                  BEQ     SOPZA
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                        ;           WHICH FOLLOWS W/ 771            <====
003040  012742  000062                          MOV     #62,-(R2)       ;MOVE TO MAILBOX #  *******  62  *******
003044  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003046  000000                                  HALT                    ;CLR INST DID NOT SET Z-BIT
003050  005300                          SOPZA:  DEC     R0              ;RESET R0
003052  005300                                  DEC     R0
003054  005120                                  COM     (R0)+           ;TRY COMPLEMENTING WITH MODE 2
003056  100004                                  BPL     SOP2B
003060  005300                                  DEC     R0              ;RESET R0
003062  005300                                  DEC     R0
003064  005220                                  INC     (R0)+           ;TRY INCREMENTING WITH MODE 2
003066  001404                                  BEQ     TST44
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                        ;           WHICH FOLLOWS W/ 755            <====
003070                                  SOP2B:
003070  012742  000063                          MOV     #63,-(R2)       ;MOVE TO MAILBOX #  *******  63  *******
003074  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003076  000000                                  HALT                    ;CHECK CUMMULATIVE RESULT OF ABOVE INST
                                                                        ;  OR SEQUENCE ERROR
```

```
1577
1578                            ;*****************************************************************************
1579                            ;
1580                            ;       THIS TEST VERIFIES MODE 2 SINGLE OPERAND INSTRUCTIONS WHICH
1581                            ;ADDRESS EVEN BYTES.  R0 IS SET TO 400 AND USED TO INITIALIZE LOCATION
1582                            ;400 TO -1.  CLRB INSTRUCTION IS THEN EXECUTED ON BYTE 400 WITH
1583                            ;MODE 2.
1584                            ;       R0 IS THEN DECREMENTED BEFORE EACH OF SEVERAL MODE 2 INSTRUCTIONS
1585                            ;WHICH ARE USED TO VERIFY THE DATA RESULTS OF THE TEST.  THIS PROCEDURE ALSO
1586                            ;VERIFIES THE PROPER INCREMENTING OF THE REGISTER.
1587                            ;
1588                            ;*****************************************************************************
1589                            ;TEST 44        TEST MODE 2 EVEN BYTE USING SOP INST.
1590                            ;*****************************************************************************
1591  003100  005212            TST44:  INC     (R2)            ;UPDATE TEST NUMBER
1592  003102  022712  000044            CMP     #44,(R2)        ;SEQUENCE ERROR?
1593  003106  001022                    BNE     TST45-10        ;BR TO ERROR HALT ON SEQ ERROR
1594  003110  005000                    CLR     R0              ;SET R0=400
1595  003112  105100                    COMB    R0
1596  003114  005200                    INC     R0
1597  003116  005010                    CLR     (R0)            ;CLEAR 400
1598  003120  005110                    COM     (R0)            ;INITIALIZE: 400=-1
1599  003122  105020                    CLRB    (R0)+           ;TRY TO CLEAT 400 W/MODE 2
1600  003124  001404                    BEQ     SOPB2A
1601                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1602                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1603                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1604                                                            ;           WHICH FOLLOWS W/ 771           <====
1605  003126  012742  000064            MOV     #64,-(R2)       ;MOVE TO MAILBOX #  *******  64  *******
1606  003132  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1607  003134  000000                    HALT                    ;CLR DID NOT SET Z-BIT
1608  003136  005300            SOPB2A: DEC     R0              ;RESULT R0=400
1609  003140  005210                    INC     (R0)            ;INC 400 TO TEST WORD
1610  003142  105420                    NEGB    (R0)+           ;TRY TO NEGATE EVEN BYTE
1611  003144  100003                    BPL     SOPB2B
1612  003146  005300                    DEC     R0              ;RESET R0=400
1613  003150  105220                    INCB    (R0)+           ;TRY INCREMENT OF EVEN BYTE
1614  003152  001404                    BEQ     TST45
1615                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1616                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1617                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1618                                                            ;           WHICH FOLLOWS W/ 756           <====
1619  003154                    SOPB2B:
1620  003154  012742  000065            MOV     #65,-(R2)       ;MOVE TO MAILBOX #  *******  65  *******
1621  003160  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1622  003162  000000                    HALT                    ;TEST CUMMULATIVE RESULT OF ABOVE INST.
1623                                                            ;  OR SEQUENCE ERROR
```

```
                                        ;****************************************************************************
                                        ;
                                        ;       THIS TEST FOLLOWS THE SAME PROCEDURE DESCRIBED IN THE PREVIOUS
                                        ;TEST.  HERE. THE BYTE INSTRUCTION IS USED TO ADDRESS AN ODD BYTE.
                                        ;
                                        ;****************************************************************************
                                        ;TEST 45        TEST MODE 2 ODD BYTE USING SOP INST.
                                        ;****************************************************************************
003164  005212                          TST45:  INC     (R2)            ;UPDATE TEST NUMBER
003166  022712  000045                          CMP     #45,(R2)        ;SEQUENCE ERROR?
003172  001025                                  BNE     TST46-10        ;BR TO ERROR HALT ON SEQ ERROR
003174  005000                                  CLR     R0              ;SET R0=400
003176  105100                                  COMB    R0
003200  005200                                  INC     R0
003202  005010                                  CLR     (R0)            ;CLEAR LOC 400
003204  005110                                  COM     (R0)            ;INITIALIZE: 400=-1
003206  005200                                  INC     R0              ;R0=ODD BYTE
003210  105020                                  CLRB    (R0)+           ;TRY TO CLEAR ODD BYTE
003212  001404                                  BEQ     SOPB2C
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS     <====
                                                                        ;           CONDITIONAL BRANCH INST. AND     <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION     <====
                                                                        ;           WHICH FOLLOWS W/ 770             <====
003214  012742  000066                          MOV     #66,-(R2)       ;MOVE TO MAILBOX #  *******  66  *******
003220  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003222  000000                                  HALT                    ;CLRB DID NOT SET Z-BIT
003224  005300                          SOPB2C: DEC     R0              ;R0=WORD ADDR.
003226  005300                                  DEC     R0
003230  005220                                  INC     (R0)+           ;INCREMENT WORD
003232  005300                                  DEC     R0              ;POINT TO ODD BYTE
003234  105420                                  NEGB    (R0)+           ;TRY TO NEGATE ODD BYTE
003236  100003                                  BPL     SOPB2D
003240  005300                                  DEC     R0              ;RESET R0 TO ODD BYTE
003242  105220                                  INCB    (R0)+           ;TRY TO INCREMENT ODD BYTE
003244  001404                                  BEQ     TST46
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS     <====
                                                                        ;           CONDITIONAL BRANCH INST. AND     <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION     <====
                                                                        ;           WHICH FOLLOWS W/ 753             <====
003246                                  SOPB2D:
003246  012742  000067                          MOV     #67,-(R2)       ;MOVE TO MAILBOX #  *******  67  *******
003252  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003254  000000                                  HALT                    ;TEST CUMMULATIVE RESULT OF ABOVE INST.
                                                                        ;  OR SEQUENCE ERROR
```

```
                                   ;*****************************************************************************
                                   ;
                                   ;        THIS TEST VERIFIES MODE 3 SINGLE OPERAND INSTRUCTIONS.  IT
                                   ;USES LOCATION 0 AS ITS TARGET DATA.  A TABLE LOCATED AT LOC. 400
                                   ;THRU 402 IS USED TO SUPPLY THE ADDRESS OF LOCATION 0 TO THE
                                   ;INSTRUCTIONS UNDER TEST.
                                   ;        R0 IS SET TO 400, THE START OF THE ADDRESS TABLE, AND A CLR
                                   ;INSTRUCTION IS EXECUTED WITH MODE 3 TO CLEAR LOC. 0.  THEN R0
                                   ;IS DECREMENTED BY TWO AND TWO OTHER MODE 3 INSTRUCTIONS OPERATE ON
                                   ;LOC. 0 TO VERIFY THE DATA RESULTS OF THE TEST.  THE PROPER INCREMENTING
                                   ;OF THE REGISTER IS ALSO VERIFIED IN THIS MANNER.
                                   ;        IF A FAILURE IS DETECTED BE SURE TO VERIFY THAT THE TABLE
                                   ;(LOC. 400-402) HAS THE PROPER VALUES (0).
                                   ;
                                   ;*****************************************************************************
                                   ;TEST 46        TEST MODE 3 USING SOP INST.
                                   ;*****************************************************************************
003256  005212                     TST46:  INC     (R2)                    ;UPDATE TEST NUMBER
003260  022712  000046                     CMP     #46,(R2)                ;SEQUENCE ERROR?
003264  001020                             BNE     TST47-10                ;BR TO ERROR HALT ON SEQ ERROR
003266  005000                             CLR     R0                      ;SET R0=400
003270  105100                             COMB    R0
003272  005200                             INC     R0
003274  005010                             CLR     (R0)                    ;CLEAR LOC 400
003276  005030                             CLR     @(R0)+                  ;TRY TO CLEAR LOC 0 USING MODE 3
003300  001404                             BEQ     SOP3A
                                                                           ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                           ;           CONDITIONAL BRANCH INST. AND   <====
                                                                           ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                           ;           WHICH FOLLOWS W/ 772           <====
003302  012742  000070                     MOV     #70,-(R2)               ;MOVE TO MAILBOX #  *******  70  *******
003306  005242                             INC     -(R2)                   ;SET MSGTYP TO FATAL ERROR
003310  000000                             HALT                            ;CLR DID NOT SET Z-BIT
003312  005300                     SOP3A:  DEC     R0                      ;RESET R0=400
003314  005300                             DEC     R0
003316  005130                             COM     @(R0)+                  ;TRY TO COMPLEMENT LOC 0 OF MODE 3
003320  100002                             BPL     SOP3B
003322  005230                             INC     @(R0)+                  ;TRY TO INCREMENT LOC 0 W/MODE 3
003324  001404                             BEQ     TST47
                                                                           ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                           ;           CONDITIONAL BRANCH INST. AND   <====
                                                                           ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                           ;           WHICH FOLLOWS W/ 760           <====
003326                             SOP3B:
003326  012742  000071                     MOV     #71,-(R2)               ;MOVE TO MAILBOX #  *******  71  *******
003332  005242                             INC     -(R2)                   ;SET MSGTYP TO FATAL ERROR
003334  000000                             HALT                            ;CUMMULATIVE RESULT OF ABOVE INST FAILED
                                                                           ;  OR SEQUENCE ERROR
```

```
1718
1719                                    ;*****************************************************************************
1720                                    ;
1721                                    ;       THIS TEST VERIFIES MODE 3 SINGLE OPERAND BYTE INSTRUCTIONS
1722                                    ;WHICH ADDRESS EVEN BYTES.  AGAIN, THE TARGET LOCATION 0 IS USED
1723                                    ;AND THE SAME TABLE AT 400 IS EMPLOYED.
1724                                    ;       AFTER POINTING R4 TO THE TABLE (400) AND SETTING LOCATION
1725                                    ;0 TO -1, A CLRB INSTRUCTION IS USED TO CLEAR BYTE 0.
1726                                    ;       SEVERAL OTHER MODE 3 INSTRUCTIONS ARE THEN USED WITH THE TABLE
1727                                    ;TO VERIFY THE DATA RESULTS AND THE PROPER INCREMENTING OF THE REGISTER.
1728                                    ;IF A FAILURE IS DETECTED, BE SURE THAT THE TABLE (LOCATION 400-402) HAS
1729                                    ;THE PROPER VALUES (0).
1730                                    ;
1731                                    ;*****************************************************************************
1732                                    ;TEST 47        TEST MODE 3 EVEN BYTE USING SOP INST.
1733                                    ;*****************************************************************************
1734  003336  005212                    TST47:  INC     (R2)            ;UPDATE TEST NUMBER
1735  003340  022712  000047                    CMP     #47,(R2)        ;SEQUENCE ERROR?
1736  003344  001026                            BNE     TST50-10        ;BR TO ERROR HALT ON SEQ ERROR
1737  003346  005004                            CLR     R4              ;SET R4=400
1738  003350  105104                            COMB    R4
1739  003352  005204                            INC     R4
1740  003354  005000                            CLR     R0              ;INITIALIZE LOC. 0=-1
1741  003356  005010                            CLR     (R0)
1742  003360  005110                            COM     (R0)
1743  003362  105034                            CLRB    @(R4)+          ;TRY TO CLEAR EVEN BYTE
1744  003364  001404                            BEQ     SOPB3A
1745                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1746                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
1747                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
1748                                                                    ;           WHICH FOLLOWS W/ 770           <====
1749  003366  012742  000072                            MOV     #72,-(R2)       ;MOVE TO MAILBOX #  *******  72  *******
1750  003372  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1751  003374  000000                            HALT                    ;CLRB DID NOT SET Z-BIT
1752  003376  005304                    SOPB3A: DEC     R4              ;RESET POINTER
1753  003400  005304                            DEC     R4
1754  003402  005234                            INC     @(R4)+          ;TRY INCREMENTING WORD
1755  003404  100006                            BPL     SOPB3B
1756  003406  105434                            NEGB    @(R4)+          ;TRY TO NEGATE EVEN BYTE
1757  003410  100004                            BPL     SOPB3B
1758  003412  005304                            DEC     R4
1759  003414  005304                            DEC     R4
1760  003416  105234                            INCB    @(R4)+          ;TRY TO INCREMENT EVEN BYTE
1761  003420  001404                            BEQ     TST50
1762                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1763                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
1764                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
1765                                                                    ;           WHICH FOLLOWS W/ 752           <====
1766  003422                            SOPB3B:
1767  003422  012742  000073                            MOV     #73,-(R2)       ;MOVE TO MAILBOX #  *******  73  *******
1768  003426  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1769  003430  000000                            HALT                    ;CUMMULATIVE RESULT OF ABOVE INST FAILED
1770                                                                    ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 3 SINGLE OPERAND BYTE INSTRUCTIONS
                                ;WHICH ADDRESS ODD BYTES.  THE TARGET IS BYTE 1.  A TABLE AT
                                ;LOC. 400-406 IS USED.  R0 SERVES AS THE TABLE POINTER.
                                ;       R0 IS INITIALIZED TO 400.  LOC. 0 IS SET TO -1 USING THE
                                ;FIRST TWO TABLE ENTRIES.  A CLRB MODE 3 IS EXECUTED ON BYTE 1 USING
                                ;TABLE ADDRESS AT 404.  R0 IS DECREMENTED TO 402 AND SEVERAL SOP
                                ;MODE 3 INSTRUCTIONS ARE USED TO VERIFY DATA RESULTS AND PROPER
                                ;REGISTER INCREMENTING.
                                ;       THE TABLE (400-406) SHOULD CONTAIN 0,0,1,1 BEFORE AND
                                ;AFTER THE TEST IS RUN.
                                ;
                                ;*****************************************************************************
                                ;TEST 50        TEST MODE 3 ODD BYTE USING SOP INST.
                                ;*****************************************************************************
003432  005212                  TST50:  INC     (R2)            ;UPDATE TEST NUMBER
003434  022712  000050                  CMP     #50,(R2)        ;SEQUENCE ERROR?
003440  001024                          BNE     TST51-10        ;BR TO ERROR HALT ON SEQ ERROR
003442  005000                          CLR     R0              ;SET R0=400
003444  105100                          COMB    R0
003446  005200                          INC     R0
003450  005030                          CLR     @(R0)+          ;INITIALIZE
003452  005130                          COM     @(R0)+          ;LOC 0=-1 R0=404
003454  105030                          CLRB    @(R0)+          ;TRY TO CLEAR ODD BYTE
003456  001404                          BEQ     SOPB3C
                                                                ;  TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;            CONDITIONAL BRANCH INST. AND   <====
                                                                ;            REPLACE THE MOVE INSTRUCTION   <====
                                                                ;            WHICH FOLLOWS W/ 771           <====
003460  012742  000074                          MOV     #74,-(R2)       ;MOVE TO MAILBOX #  *******  74  *******
003464  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003466  000000                          HALT                    ;CLRB DID NOT SET Z-BIT
003470  005300                  SOPB3C: DEC     R0              ;RESET R0
003472  005300                          DEC     R0
003474  005300                          DEC     R0              ;POINT TO EVEN BYTE ADDR.
003476  005300                          DEC     R0
003500  005230                          INC     @(R0)+          ;INCREMENT WORD
003502  105430                          NEGB    @(R0)+          ;TRY TO NEGATE ODD BYTE
003504  100002                          BPL     SOPB3D
003506  10523C                          INCB    @(R0)+          ;TRY TO INCREMENT ODD BYTE
003510  001404                          BEQ     TST51
                                                                ;  TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;            CONDITIONAL BRANCH INST. AND   <====
                                                                ;            REPLACE THE MOVE INSTRUCTION   <====
                                                                ;            WHICH FOLLOWS W/ 754           <====
003512                  SOPB3D:
003512  012742  000075                          MOV     #75,-(R2)       ;MOVE TO MAILBOX #  *******  75  *******
003516  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003520  000000                          HALT                    ;CUMMULATIVE RESULT OF ABOVE INSTS FAILED
                                                                ;  OR SEQUENCE ERROR
```

```
1823
1824                                    ;*****************************************************************************
1825                                    ;TEST 51        TEST MODE 4 USING SOP INSTS
1826                                    ;*****************************************************************************
1827  003522  005212                    TST51:  INC     (R2)            ;UPDATE TEST NUMBER
1828  003524  022712  000051                    CMP     #51,(R2)        ;SEQUENCE ERROR?
1829  003530  001021                            BNE     TST52-10        ;BR TO ERROR HALT ON SEQ ERROR
1830  003532  005000                            CLR     R0              ;SET R0=400
1831  003534  105100                            COMB    R0
1832  003536  005200                            INC     R0
1833  003540  005040                            CLR     -(R0)           ;TRY TO CLEAR USING MODE 4
1834  003542  001404                            BEQ     SOP4A
1835                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1836                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
1837                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
1838                                                                    ;           WHICH FOLLOWS W/ 773           <====
1839  003544  012742  000076                  MOV     #76,-(R2)       ;MOVE TO MAILBOX #  *******  76  *******
1840  003550  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1841  003552  000000                            HALT                    ;CLR DID NOT SET Z-BIT
1842  003554  005200                    SOP4A:  INC     R0              ;RESET R0
1843  003556  005200                            INC     R0
1844  003560  005140                            COM     -(R0)           ;TRY TO COMPLEMENT USING MODE 4
1845  003562  100004                            BPL     SOP4B
1846  003564  005200                            INC     R0              ;MOVE POINTER
1847  003566  005200                            INC     R0
1848  003570  005240                            INC     -(R0)
1849  003572  001404                            BEQ     TST52
1850                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1851                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
1852                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
1853                                                                    ;           WHICH FOLLOWS W/ 757           <====
1854  003574                            SOP4B:
1855  003574  012742  000077                  MOV     #77,-(R2)       ;MOVE TO MAILBOX #  *******  77  *******
1856  003600  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1857  003602  000000                            HALT                    ;CHECK CUMMULATIVE RESULT OF ABOVE INST.
1858                                                                    ;  OR SEQUENCE ERROR
```

```
                                  ;*************************************************************************
                                  ;
                                  ;       THIS TEST VERIFIES MODE 5 SINGLE OPERAND INSTRUCTIONS.  IT
                                  ;USES LOCATION 0 AS ITS TARGET DATA.  A TABLE LOCATED AT LOC. 372
                                  ;THRU 374 IS USED TO SUPPLY THE ADDRESS OF LOCATION 0 TO THE
                                  ;INSTRUCTIONS UNDER TEST
                                  ;       R0 IS SET TO 376, (THE START OF THE ADDRESS TABLE) +2,
                                  ;AND A CLR INSTRUCTION IS EXECUTED WITH MODE 3 TO CLEAR
                                  ;LOC. 0.  THEN R0 IS INCREMENTED BY TWO AND TWO OTHER MODE 3
                                  ;INSTRUCTIONS OPERATE ON LOC. 0 TO VERIFY THE DATA RESULTS OF
                                  ;THE TEST.  THE PROPER DECREMENTING OF THE REGISTER IS ALSO
                                  ;VERIFIED IN THIS MANNER.
                                  ;       IF A FAILURE IS DETECTED BE SURE TO VERIFY THAT THE TABLE
                                  ;(LOC. 372 THRU 374) HAS THE PROPER VALUES (0).
                                  ;
                                  ;*************************************************************************
                                  ;TEST 52        TEST MODE 5 USING SOP INSTS
                                  ;*************************************************************************
003604  005212                    TST52:  INC     (R2)            ;UPDATE TEST NUMBER
003606  022712  000052                    CMP     #52,(R2)        ;SEQUENCE ERROR?
003612  001017                            BNE     TST53-10        ;BR TO ERROR HALT ON SEQ ERROR
003614  005000                            CLR     R0              ;SET R0=376
003616  005020                            CLR     (R0)+
003620  105400                            NEGB    R0
003622  005050                            CLR     @-(R0)          ;TRY TO CLEAR LOC 0 W/MODE 5
003624  001404                            BEQ     SOP5A
                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS     <====
                                                                  ;           CONDITIONAL BRANCH INST. AND     <====
                                                                  ;           REPLACE THE MOVE INSTRUCTION     <====
                                                                  ;           WHICH FOLLOWS W/ 773             <====
003626  012742  000100                    MOV     #100,-(R2)      ;MOVE TO MAILBOX #  *******  100  *******
003632  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003634  000000                            HALT                    ;CLR DID NOT SET Z-BIT
003636  005200                    SOP5A:  INC     R0              ;RESET R0
003640  005200                            INC     R0
003642  005150                            COM     @-(R0)          ;TRY TO COMPLEMENT LOC. 0 W/MODE 5
003644  100002                            BPL     SOP5B
003646  005250                            INC     @-(R0)          ;TRY TO INCREMENT LOC. 0 W/MODE 5
003650  001404                            BEQ     TST53
                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS     <====
                                                                  ;           CONDITIONAL BRANCH INST. AND     <====
                                                                  ;           REPLACE THE MOVE INSTRUCTION     <====
                                                                  ;           WHICH FOLLOWS W/ 761             <====
003652                            SOP5B:
003652  012742  000101                    MOV     #101,-(R2)      ;MOVE TO MAILBOX #  *******  101  *******
003656  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003660  000000                            HALT                    ;TEST CUMMULATIVE RESULT OF ABOVE INSTS
                                                                  ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 6 SINGLE OPERAND INSTRUCTIONS.  IT
                                ;USES LOCATION 0 AS ITS TARGET DATA.  R0 IS SET TO 400 USING
                                ;PREVIOUSLY TESTED INSTRUCTIONS AND A MODE 6 CLR INSTRUCTION IS
                                ;EXECUTED ON LOC. 0 USING R0 AND A -400 OFFSET.  COM AND INC
                                ;INSTRUCTIONS ARE THEN USED TO VERIFY THE DATA.
                                ;
                                ;*****************************************************************************
                                ;TEST 53         TEST MODE 6 USING SOP INSTS
                                ;*****************************************************************************
003662  005212                TST53:  INC     (R2)            ;UPDATE TEST NUMBER
003664  022712  000053                CMP     #53,(R2)        ;SEQUENCE ERROR?
003670  001020                        BNE     TST54-10        ;BR TO ERROR HALT ON SEQ ERROR
003672  005000                        CLR     R0              ;SET R0=400
003674  105100                        COMB    R0
003676  005200                        INC     R0
003700  005060  177400                CLR     -400(R0)        ;TRY TO CLEAR LOCATION 0 W/MODE 6
003704  001404                        BEQ     SOP6A
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 772           <====
003706  012742  000102                MOV     #102,-(R2)      ;MOVE TO MAILBOX #  *******  102  *******
003712  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003714  000000                        HALT                    ;CLR DID NOT SET Z-BIT
003716  005160  177400        SOP6A:  COM     -400(R0)        ;TRY TO COMPLEMENT LOCATION 0 W/MODE 6
003722  100003                        BPL     SOP6B
003724  005260  177400                INC     -400(R0)        ;TRY TO INCREMENT LOCATION 0 W/MODE 6
003730  001404                        BEQ     TST54
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 760           <====
003732                          SOP6B:
003732  012742  000103                MOV     #103,-(R2)      ;MOVE TO MAILBOX #  *******  103  *******
003736  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
003740  000000                        HALT                    ;TEST CUMMULATIVE RESULT OF ABOVE INSTS
                                                            ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 7 SINGLE OPERAND INSTRUCTIONS.  IT USES
                                ;THE POINTER TO LOC. 0 WHICH IS STORED AT LOC. 402.
                                ;       R0 IS SET TO 400 AND A MODE 7 CLR INSTRUCTION IS
                                ;EXECUTED WITH A +2 OFFSET TO CLEAR LOC. 0.
                                ;       SEVERAL OTHER MODE 7 INSTRUCTIONS ARE THEN USED ON THE COMMON
                                ;LOCATION TO VERIFY THE DATA RESULTS.
                                ;
                                ;*****************************************************************************
                                ;TEST 54         TEST MODE 7 USING SOP INST.
                                ;*****************************************************************************
1961  003742  005212            ;ST54:  INC     (R2)            ;UPDATE TEST NUMBER
1962  003744  022712  000054            CMP     #54,(R2)        ;SEQUENCE ERROR?
1963  003750  001020                    BNE     TST55-10        ;BR TO ERROR HALT ON SEQ ERROR
1964  003752  005000                    CLR     R0              ;SET R0=400
1965  003754  105100                    COMB    R0
1966  003756  005200                    INC     R0
1967  003760  005070  000002            CLR     @2(R0)          ;TRY TO CLEAR LOC. 0 W/MODE 7
1968  003764  001404                    BEQ     SOP7A
1969                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1970                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1971                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1972                                                            ;           WHICH FOLLOWS W/ 772           <====
1973  003766  012742  000104            MOV     #104,-(R2)      ;MOVE TO MAILBOX #  *******  104  *******
1974  003772  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1975  003774  000000                    HALT                    ;CLR DID NOT SET Z-BIT
1976  003776  005170  000002    SOP7A:  COM     @2(R0)          ;TRY TO COMPLEMENT LOC. 0 W/MODE 7
1977  004002  100003                    BPL     SOP7B
1978  004004  005270  000002            INC     @2(R0)          ;TRY TO INCREMENT LOC. 0 W/MODE 7
1979  004010  001404                    BEQ     TST55
1980                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
1981                                                            ;           CONDITIONAL BRANCH INST. AND   <====
1982                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
1983                                                            ;           WHICH FOLLOWS W/ 760           <====
1984  004012            SOP7B:
1985  004012  012742  000105            MOV     #105,-(R2)      ;MOVE TO MAILBOX #  *******  105  *******
1986  004016  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
1987  004020  000000                    HALT                    ;TEST CUMMULATIVE RESULT OF ABOVE INSTS.
1988                                                            ;  OR SEQUENCE ERROR
```

C04

```
                                        ;*************************************************************************
                                        ;
                                        ;        THIS TEST VERIFIES PROGRAM COUNTER ADDRESSING WITH SOP
                                        ;INSTRUCTIONS.  CLR MODE 77 IS USED TO CLEAR THE LOCATION FOLLOWING THE
                                        ;INSTRUCTION (SOPX).  THEN SINGLE OPERAND INSTRUCTIONS WITH MODES 37, 67, AND
                                        ;77, USING INDIRECT POINTER SOPXAD ARE USED TO VERIFY THE DATA RESULTS
                                        ;OF THESE INSTRUCTIONS.
                                        ;
                                        ;*************************************************************************
                                        ;TEST 55        TEST SOP INSTRUCTIONS MODES 2,3,6,7 WITH REGISTER 7
                                        ;*************************************************************************
004022  005212                          TST55:  INC     (R2)            ;UPDATE TEST NUMBER
004024  022712  000055                          CMP     #55,(R2)        ;SEQUENCE ERROR?
004030  001017                                  BNE     SOPB            ;BR TO ERROR HALT ON SEQ ERROR
004032  005027                                  CLR     (R7)+           ;CLEAR NEXT LOCATION: (SOPX)
004034  177777                          SOPX:   -1                      ;USE MODE 27
004036  001404                                  BEQ     SOPA
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                        ;           WHICH FOLLOWS W/ 775            <====
004040  012742  000106                          MOV     #106,-(R2)      ;MOVE TO MAILBOX #  *******  106  *******
004044  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004046  000000                                  HALT                    ;CLR DID NOT SET Z-BIT
004050  005237  004034                  SOPA:   INC     @#SOPX          ;INC SOPX W/MODE 37
004054  005467  177754                          NEG     SOPX            ;NEGATE SOPX W/MODE 67
004060  100003                                  BPL     SOPB
004062  005277  000012                          INC     @SOPXAD         ;INC SOPX W/MODE 77
004066  001405                                  BEQ     TST56
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                        ;           WHICH FOLLOWS W/ 761            <====
004070                                  SOPB:
004070  012742  000107                          MOV     #107,-(R2)      ;MOVE TO MAILBOX #  *******  107  *******
004074  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004076  000000                                  HALT                    ;INC DID NOT SET Z-BIT
                                                                        ;  OR SEQUENCE ERROR
004100  004034                          SOPXAD: SOPX                    ;INDIRECT ADDRESS OF SOPX
```

```
                                ;*******************************************************************
                                ;
                                ;       THIS TEST VERIFIES SINGLE OPERAND NON-MODIFYING INSTRUCTIONS
                                ;USING MODE 0.  R0 IS SET TO ZERO AND THE CONDITION CODES ARE SET
                                ;TO THE COMPLEMENT OF THAT EXPECTED BY THE INSTRUCTION.  A TST INSTRUCTION
                                ;IS EXECUTED AND CONDITIONAL BRANCHES ARE USED TO TEST THE CONDITION
                                ;CODES.
                                ;
                                ;*******************************************************************
                                ;TEST 56        TEST MODE 0 SOP NON-MODIFYING
                                ;*******************************************************************
004102  005212                  TST56:  INC     (R2)            ;UPDATE TEST NUMBER
004104  022712  000056                  CMP     #56,(R2)        ;SEQUENCE ERROR?
004110  001010                          BNE     TST57-10        ;BR TO ERROR HALT ON SEQ ERROR
004112  005000                          CLR     R0              ;INITIALIZE R0=0
004114  000277                          SCC                     ;SET CC=1011
004116  000244                          CLZ
004120  005700                          TST     R0              ;TRY TST W/ MODE 0
004122  102403                          BVS     SNMOA           ;CHECK THAT CC=0100
004124  100402                          BMI     SNMOA
004126  103401                          BCS     SNMOA
004130  001404                          BEQ     TST57
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                ;          CONDITIONAL BRANCH INST. AND    <====
                                                                ;          REPLACE THE MOVE INSTRUCTION     ====
                                                                ;          WHICH FOLLOWS W/ 770            <====
004132                          SNMOA:
004132  012742  000110                  MOV     #110,-(R2)      ;MOVE TO MAILBOX #  *******  110  *******
004136  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004140  000000                          HALT                    ;CONDITION CODES NOT SET PROPERLY
                                                                ;  OR SEQUENCE ERROR
```

```
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073  004142  005212                  TST57:  INC     (R2)            ;UPDATE TEST NUMBER
2074  004144  022712  000057                  CMP     #57,(R2)        ;SEQUENCE ERROR?
2075  004150  001010                          BNE     TST60-10        ;BR TO ERROR HALT ON SEQ ERROR
2076  004152  005000                          CLR     R0              ;INITIALIZE
2077  004154  105100                          COMB    R0              ;R0=377
2078  004156  000277                          SCC                     ;SET CC=0111
2079  004160  000250                          CLN
2080  004162  105700                          TSTB    R0              ;TRY TST EVEN BYTE
2081  004164  102402                          BVS     SNMBOA          ;CHECK CC=1000
2082  004166  101401                          BLOS    SNMBOA
2083  004170  100404                          BMI     TST60
2084                                                                   ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2085                                                                   ;           CONDITIONAL BRANCH INST. AND   <====
2086                                                                   ;           REPLACE THE MOVE INSTRUCTION   <====
2087                                                                   ;           WHICH FOLLOWS W/ 770           <====
2088  004172                          SNMBOA:
2089  004172  012742  000111                  MOV     #111,-(R2)      ;MOVE TO MAILBOX #  *******  111  *******
2090  004176  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2091  004200  000000                          HALT                    ;CONDITION CODES NOT SET PROPERLY
2092                                                                   ;  OR SEQUENCE ERROR
```

```
                                ;*************************************************************************
                                ;
                                ;       THIS TEST VERIFIES SINGLE OPERAND INSTRUCTIONS WITH MODE 1.
                                ;RO IS USED TO POINT TO AND CLEAR LOC. 0.  THE COMPLEMENT OF THE
                                ;EXPECTED CONDITION CODES ARE LOADED IN THE PSW.  A TST INSTRUCTION
                                ;IS THEN EXECUTED ON LOC. 0 USING R0 AND CONDITIONAL BRANCHES TEST
                                ;THE RESULTS.
                                ;
                                ;*************************************************************************
                                ;TEST 60        TEST MODE 1 SOP NON-MODIFYING
                                ;*************************************************************************
004202  005212                  TST60:  INC     (R2)            ;UPDATE TEST NUMBER
004204  022712  000060                  CMP     #60,(R2)        ;SEQUENCE ERROR?
004210  001011                          BNE     TST61-10        ;BR TO ERROR HALT ON SEQ ERROR
004212  005000                          CLR     R0              ;POINT TO LOC 0
004214  005010                          CLR     (R0)            ;CLEAR LOC 0
004216  000277                          SCC                     ;INITIALIZE
004220  000244                          CLZ                     ;CC=1011
004222  005710                          TST     (R0)            ;TRY TST W/ MODE 1
004224  102403                          BVS     SNM1A           ;CHECK CC=0100
004226  103402                          BCS     SNM1A
004230  100401                          BMI     SNM1A
004232  001404                          BEQ     TST61
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                ;           CONDITIONAL BRANCH INST. AND    <====
                                                                ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                ;           WHICH FOLLOWS W/ 767            <====
004234                          SNM1A:
004234  012742  000112                  MOV     #112,-(R2)      ;MOVE TO MAILBOX #  *******  112  *******
004240  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004242  000000                          HALT                    ;CC'S NOT SET PROPERLY
                                                                ;  OR SEQUENCE ERROR
```

```
                                  ;*************************************************************************
                                  ;
                                  ;        THIS TEST SETS LOCATION 0 TO 377 AND THEN USES R0 TO TEST
                                  ;THE EVEN BYTE AND THE ODD BYTE USING SOP BYTE INSTRUCTIONS WITH MODE 1.
                                  ;AGAIN, CONDITIONAL BRANCHES ARE USED TO VERIFY THE SETTING OF THE
                                  ;PROPER CONDITION CODE BITS.
                                  ;
                                  ;*************************************************************************
                                  ;TEST 61         TEST MODE 1 BYTE INST. NON-MODIFYING
                                  ;*************************************************************************
004244  005212                    TST61:  INC     (R2)            ;UPDATE TEST NUMBER
004246  022712  000061                    CMP     #61,(R2)        ;SEQUENCE ERROR?
004252  001026                            BNE     TST62-10        ;BR TO ERROR HALT ON SEQ ERROR
004254  005000                            CLR     R0              ;POINT TO LOC 0
004256  005010                            CLR     (R0)            ;CLEAR LOC 0
004260  105110                            COMB    (R0)            ;COMPLEMENT BYTE 0
004262  000277                            SCC                     ;SET CC=0111
004264  000250                            CLN
004266  105710                            TSTB    (R0)            ;TRY TST ON EVEN BYTE
004270  102402                            BVS     SNMB1A
004272  101401                            BLOS    SNMB1A
004274  100404                            BMI     SNMB1B
                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                  ;           CONDITIONAL BRANCH INST. AND     <====
                                                                  ;           REPLACE THE MOVE INSTRUCTION     <====
                                                                  ;           WHICH FOLLOWS W/ 767             <====
004276                            SNMB1A:
004276  012742  000113                    MOV     #113,-(R2)      ;MOVE TO MAILBOX #  *******  113  *******
004302  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004304  000000                            HALT                    ;CC'S NOT CORRECT
004306  005000                    SNMB1B: CLR     R0
004310  005200                            INC     R0
004312  000277                            SCC                     ;SET CC=1011
004314  000244                            CLZ
004316  105710                            TSTB    (R0)            ;TRY TO TST AN ODD BYTE
004320  102403                            BVS     SNMB1C          ;CHECK CC=0100
004322  103402                            BCS     SNMB1C
004324  100401                            BMI     SNMB1C
004326  001404                            BEQ     TST62
                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                  ;           CONDITIONAL BRANCH INST. AND     <====
                                                                  ;           REPLACE THE MOVE INSTRUCTION     <====
                                                                  ;           WHICH FOLLOWS W/ 752             <====
004330                            SNMB1C:
004330  012742  000114                    MOV     #114,-(R2)      ;MOVE TO MAILBOX #  *******  114  *******
004234  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004336  000000                            HALT                    ;CC'S NOT CORRECT
                                                                  ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES THE SINGLE-OPERAND NON-MODIFYING INSTRUCTIONS
                                ;USING MODE 2.  IT USES THE IDENTICAL PROCEDURE EMPLOYED IN THE
                                ;MODE 1 TESTS.  ADDITIONALLY, THE REGISTER IS CHECKED TO ASSURE THAT
                                ;IT IS INCREMENTED PROPERLY.
                                ;
                                ;*****************************************************************************
                                ;TEST 62         TEST MODE 2 WITH SOP NON-MODIFYING
                                ;*****************************************************************************
004340  005212                  TST62:  INC     (R2)            ;UPDATE TEST NUMBER
004342  022712  000062                  CMP     #62,(R2)        ;SEQUENCE ERROR?
004346  001020                          BNE     TST63-10        ;BR TO ERROR HALT ON SEQ ERROR
004350  005000                          CLR     R0              ;INITIALIZE R0=0
004352  005010                          CLR     (R0)            ;CLEAR LOC 0
004354  000277                          SCC                     ;SET CC=1011
004356  000244                          CLZ
004360  005720                          TST     (R0)+           ;TRY TST W/ MODE 2
004362  102403                          BVS     SNM2A           ;CHECK CC=0100
004364  103402                          BCS     SNM2A
004366  100401                          BMI     SNM2A
004370  001404                          BEQ     SNM2B
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 767           <====
004372                          SNM2A:
004372  012742  000115                  MOV     #115,-(R2)      ;MOVE TO MAILBOX #  *******  115  *******
004376  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004400  000000                          HALT                    ;CC'S NOT CORRECT
004402  005300                  SNM2B:  DEC     R0              ;RESET R0
004404  005300                          DEC     R0
004406  001404                          BEQ     TST63
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 760           <====
004410  012742  000116                  MOV     #116,-(R2)      ;MOVE TO MAILBOX #  *******  116  *******
004414  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004416  000000                          HALT                    ;MODE 2 DID NOT INC REQ CORRECTLY
                                                                ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 2 SINGLE OPERAND NON-MODIFYING BYTE
                                ;INSTRUCTIONS IT USES R0 TO POINT TO LOC. 0.  WITH LOCATION 0
                                ;SET TO 377, THE EVEN AND ODD BYTE IS TESTED WITH TSTB INSTRUCTIONS
                                ;TO VERIFY THE CORRECT CC ARE SET.  THE REGISTER IS CHECKED FOR
                                ;PROPER INCREMENTING.
                                ;
                                ;*****************************************************************************
                                ;TEST 63         TEST MODE 2 - BYTE W/ SOP NON-MODIFYING
                                ;*****************************************************************************
004420  005212                TST63:  INC     (R2)            ;UPDATE TEST NUMBER
004422  022712  000063              CMP     #63,(R2)        ;SEQUENCE ERROR?
004426  001042                      BNE     TST64-10        ;BR TO ERROR HALT ON SEQ ERROR
004430  005000                      CLR     R0              ;CLEAR R0
004432  005010                      CLR     (R0)            ;CLEAR LOC 0
004434  105110                      COMB    (R0)            ;SET LOC 0=377
004436  000277                      SCC                     ;SET CC=0111
004440  000250                      CLN
004442  105720                      TSTB    (R0)+           ;TRY TST OF EVEN BYTE
004444  102402                      BVS     SNMB2A
004446  101401                      BLOS    SNMB2A
004450  100404                      BMI     SNMB2B
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 767           <====
004452                          SNMB2A:
004452  012742  000117                MOV     #117,-(R2)      ;MOVE TO MAILBOX #  *******  117  *******
004456  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004460  000000                      HALT                    ;CC'S NOT SET CORRECTLY
004462  005300                SNMB2B: DEC     R0              ;DECREMENT R0
004464  001404                      BEQ     SNMB2C
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 761           <====
004466  012742  000120                MOV     #120,-(R2)      ;MOVE TO MAILBOX #  *******  120  *******
004472  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004474  000000                      HALT                    ;MODE 2 DID NOT INC REG CORRECTLY
004476  005200                SNMB2C: INC     R0              ;POINT TO ODD BYTE
004500  000277                      SCC                     ;SET CC=1011
004502  000244                      CLZ
004504  105720                      TSTB    (R0)+           ;TRY TST OF ODD BYTE
004506  102403                      BVS     SNMB2D          ;CHECK CC'S=0100
004510  103402                      BCS     SNMB2D
004512  100401                      BMI     SNMB2D
004514  001404                      BEQ     SNMB2E
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 745           <====
004516                          SNMB2D:
004516  012742  000121                MOV     #121,-(R2)      ;MOVE TO MAILBOX #  *******  121  *******
004522  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
```

```
004524  000000                  HALT                    ;CC'S NOT CORRECT
004526  005300          SNMB2E: DEC     R0
004530  005300                  DEC     R0
004532  001404                  BEQ     TST64
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                        ;           CONDITIONAL BRANCH INST. AND      <====
                                                        ;           REPLACE THE MOVE INSTRUCTION      <====
                                                        ;           WHICH FOLLOWS W/ 736              <====
004534  012742  000122          MOV     #122,-(R2)      ;MOVE TO MAILBOX #  *******  122  *******
004540  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004542  000000                  HALT                    ;R0 DID NOT INCREMENT PROPERLY
                                                        ;  OR SEQUENCE ERROR
```

```
2285
2286                                    ;*****************************************************************************
2287                                    ;
2288                                    ;       THIS TEST VERIFIES MODE 3 SINGLE OPERAND NON-MODIFYING INSTRUCTIONS.
2289                                    ;A POINTER IN A TABLE AT LOC. 376 IS USED TO TEST LOCATION 0.
2290                                    ;THE CC'S AND THE REGISTER ARE CHECKED FOLLOWING THE
2291                                    ;TST MODE 3 INSTRUCTION.
2292                                    ;
2293                                    ;*****************************************************************************
2294                                    ;TEST 64        TEST MODE 3 W/ SOP NON-MODIFYING INSTS
2295                                    ;*****************************************************************************
2296  004544  005212                    TST64:  INC     (R2)            ;UPDATE TEST NUMBER
2297  004546  022712  000064                    CMP     #64,(R2)        ;SEQUENCE ERROR?
2298  004552  001022                            BNE     TST65-10        ;BR TO ERROR HALT ON SEQ ERROR
2299  004554  005000                            CLR     R0              ;R0=0
2300  004556  005010                            CLR     (R0)            ;CLEAR LOC 0
2301  004560  105100                            COMB    R0              ;R0=376
2302  004562  005300                            DEC     R0
2303  004564  000277                            SCC                     ;SET CC=1011
2304  004566  000244                            CLZ
2305  004570  005730                            TST     @(R0)+          ;TRY TST W/ MODE 3
2306  004572  102403                            BVS     SNM3A           ;CHECK CC=0100
2307  004574  103402                            BCS     SNM3A
2308  004576  100401                            BMI     SNM3A
2309  004600  001404                            BEQ     SNM3B
2310                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2311                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2312                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2313                                                                    ;           WHICH FOLLOWS W/ 765           <====
2314  004602                            SNM3A:
2315  004602  012742  000123                    MOV     #123,-(R2)      ;MOVE TO MAILBOX #  *******  123  *******
2316  004606  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2317  004610  000000                            HALT                    ;CC'S NOT CORRECT
2318  004612  005300                    SNM3B:  DEC     R0              ;R0=377
2319  004614  105100                            COMB    R0              ;R0=0
2320  004616  001404                            BEQ     TST65
2321                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2322                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2323                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2324                                                                    ;           WHICH FOLLOWS W/ 756           <====
2325  004620  012742  000124                    MOV     #124,-(R2)      ;MOVE TO MAILBOX #  *******  124  *******
2326  004624  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2327  004626  000000                            HALT                    ;MODE 3 DID NOT INC REG CORRECTLY
2328                                                                    ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES SOP NON-MODIFYING BYTE INSTRUCTIONS MODE 3
                                ;LOC. 0 IS SET TO 377.  TABLE AT LOC. 402-404 IS USED TO TEST
                                ;BYTE 0 AND BYTE 1.  THE REGISTER IS CHECKED FOR PROPER INCREMENTING AND
                                ;THE CC'S ARE VERIFIED.
                                ;       THE TABLE AT LOC. 402-404 SHOULD CONTAIN 0 AND 1 BEFORE AND
                                ;AFTER THE TEST IS RUN.
                                ;
                                ;*****************************************************************************
                                ;TEST 65        TEST MODE 3 - BYTES W/ SOP NON-MODIFYING INSTS.
                                ;*****************************************************************************
004630  005212        TST65:  INC     (R2)            ;UPDATE TEST NUMBER
004632  022712  000065        CMP     #65,(R2)        ;SEQUENCE ERROR?
004636  001036                BNE     TST66-10        ;BR TO ERROR HALT ON SEQ ERROR
004640  005000                CLR     R0              ;R0=0
004642  005010                CLR     (R0)            ;CLEAR LOC 0
004644  105110                COMB    (R0)            ;LOC. 0 =377
004646  105100                COMB    R0
004650  005200                INC     R0
004652  005720                TST     (R0)+           ;R0=402
004654  000277                SCC                     ;CC=0111
004656  000250                CLN
004660  105730                TSTB    @(R0)+          ;TRY TST OF EVEN BYTE
004662  102402                BVS     SNMB3A          ;CHECK CC=1000
004664  101401                BLOS    SNMB3A
004666  100404                BMI     SNMB3B
                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                ;           CONDITIONAL BRANCH INST. AND   <====
                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                ;           WHICH FOLLOWS W/ 764           <====
004670          SNMB3A:
004670  012742  000125        MOV     #125,-(R2)      ;MOVE TO MAILBOX #  *******  125  *******
004674  005242                INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004676  000000                HALT                    ;CC'S NOT CORRECT
004700  000277        SNMB3B: SCC                     ;SET CC=1011
004702  000244                CLZ
004704  105730                TSTB    @(R0)+          ;TRY TST OF ODD BYTE
004706  102403                BVS     SNMB3C          ;CHECK CC=0100
004710  103402                BCS     SNMB3C
004712  100401                BMI     SNMB3C
004714  001404                BEQ     SNMB3D
                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                ;           CONDITIONAL BRANCH INST. AND   <====
                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                ;           WHICH FOLLOWS W/ 751           <====
004716          SNMB3C:
004716  012742  000125        MOV     #126,-(R2)      ;MOVE TO MAILBOX #  *******  126  *******
004722  005242                INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004724  000000                HALT                    ;CC'S NOT CORRECT
004726  005720        SNMB3D: TST     (R0)+           ;R0=410
004730  005710                TST     (R0)
004732  100404                BMI     TST66
                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                ;           CONDITIONAL BRANCH INST. AND   <====
```

```
                                                        ;          REPLACE THE MOVE INSTRUCTION   <====
                                                        ;          WHICH FOLLOWS W/ 742           <====
004734  012742  000127          MOV     #127,-(R2)      ;MOVE TO MAILBOX #  *******  127  *******
004740  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
004742  000000                  HALT                    ;TSTB DID NOT INCREMENT R0 CORRETCLY
                                                        ;  OR SEQUENCE ERROR
                        ;*****************************************************************************
                        ;
                        ;       THIS TEST VERIFIES MODE 4 SOP NON-MODIFYING INSTRUCTIONS.
                        ;LOC. 0 IS SET TO -1 AND THE CC'S ARE SET TO THE COMPLEMENT OF THE
                        ;EXPECTED RESULTS.  R0 AND SET TO 2 AND A TST MODE 4 IS EXECUTED.
                        ;THE CC'S ARE CHECKED WITH CONDITIONAL BRANCH INSTRUCTIONS AND THE REGISTER
                        ;IS CHECKED FOR PROPER DECREMENTING.
                        ;
                        ;*****************************************************************************
                        ;TEST 66        TEST MODE 4 W/ SOP NON-MODIFYING INSTS
                        ;*****************************************************************************
004744  005212          TST66:  INC     (R2)            ;UPDATE TEST NUMBER
004746  022712  000066          CMP     #66,(R2)        ;SEQUENCE ERROR?
004752  001017                  BNE     TST67-10        ;BR TO ERROR HALT ON SEQ ERROR
004754  005000                  CLR     R0              ;R0=0
004756  005010                  CLR     (R0)            ;LOC 0=0
004760  005120                  COM     (R0)+           ;LOC 0=-1
004762  000277                  SCC                     ;SET CC=1011
004764  000244                  CLZ
004766  005740                  TST     -(R0)           ;TRY TST W/ MODE 4
004770  102402                  BVS     SNM4A           ;CHECK CC=0100
004772  101401                  BLOS    SNM4A
004774  100404                  BMI     SNM4B
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;          CONDITIONAL BRANCH INST. AND   <====
                                                        ;          REPLACE THE MOVE INSTRUCTION   <====
                                                        ;          WHICH FOLLOWS W/ 767           <====
004776          SNM4A:
004776  012742  000130          MOV     #130,-(R2)      ;MOVE TO MAILBOX #  *******  130  *******
005002  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
005004  000000                  HALT                    ;CC'S NOT CORRECT
005006  005700          SNM4B:  TST     R0
005010  001404                  BEQ     TST67
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;          CONDITIONAL BRANCH INST. AND   <====
                                                        ;          REPLACE THE MOVE INSTRUCTION   <====
                                                        ;          WHICH FOLLOWS W/ 761           <====
005012  012742  000131          MOV     #131,-(R2)      ;MOVE TO MAILBOX #  *******  131  *******
005016  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
005020  000000                  HALT                    ;TST MODE 4 DID NOT DEC R0 CORRECTLY
                                                        ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 5 SOP NON-MODIFYING INSTRUCTIONS.
                                ;IT USES A POINTER AT LOC. 376 TO TEST LOC. 0.  R0 IS SET
                                ;TO 400, A TST MODE 5 INSTRUCTION IS EXECUTED AND THE CC'S CHECKED.
                                ;R0 IS CHECKED TO INSURE PROPER DECREMENTING.
                                ;
                                ;*****************************************************************************
                                ;TEST 67        TEST MODE 5 W/ SOP NON-MODIFYING INSTS
                                ;*****************************************************************************
005022  005212                  TST67:  INC     (R2)            ;UPDATE TEST NUMBER
005024  022712  000067                  CMP     #67,(R2)        ;SEQUENCE ERROR?
005030  001022                          BNE     TST70-10        ;BR TO ERROR HALT ON SEQ ERROR
005032  005000                          CLR     R0              ;R0=0
005034  005010                          CLR     (R0)            ;LOC 0=0
005036  005110                          COM     (R0)            ;LOC 0=-1
005040  105100                          COMB    R0              ;R0=377
005042  005200                          INC     R0              ;R0=400
005044  000277                          SCC                     ;SET CC=0111
005046  000250                          CLN
005050  005750                          TST     @-(R0)          ;TRY TST W/ MODE 5
005052  102402                          BVS     SNM5A           ;CHECK CC=1000
005054  101401                          BLOS    SNM5A
005056  100404                          BMI     SNM5B
                                ;                               ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                ;                               ;           CONDITIONAL BRANCH INST. AND   <====
                                ;                               ;           REPLACE THE MOVE INSTRUCTION   <====
                                ;                               ;           WHICH FOLLOWS W/ 765           <====
005060                          SNM5A:
005060  012742  000132                  MOV     #132,-(R2)      ;MOVE TO MAILBOX #  *******  132  *******
005064  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
005066  000000                          HALT                    ;CC'S NOT SET PROPERLY
005070  005200                  SNM5B:  INC     R0              ;R0=377
005072  105100                          COMB    R0              ;R0=0
005074  001404                          BEQ     TST70
                                ;                               ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                ;                               ;           CONDITIONAL BRANCH INST. AND   <====
                                ;                               ;           REPLACE THE MOVE INSTRUCTION   <====
                                ;                               ;           WHICH FOLLOWS W/ 756           <====
005076  012742  000133                  MOV     #133,-(R2)      ;MOVE TO MAILBOX #  *******  133  *******
005102  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
005104  000000                          HALT                    ;MODE 5 DID NOT DEC R0 CORRECTLY
                                ;                               ;  OR SEQUENCE ERROR
```

```
                        ;*****************************************************************************
                        ;
                        ;       THIS TEST VERIFIES MODE 6 SOP NON-MODIFYING INSTRUCTIONS.
                        ;R0 IS SET TO 377 AND A MODE 6 TST INSTRUCTION IS EXECUTED
                        ;USING R0 AND AN OFFSET OF -377.  THE CC'S ARE CHECKED AS WELL
                        ;AS R0 TO INSURE IT WAS NOT ALTERED.
                        ;
                        ;*****************************************************************************
                        ;TEST 70         TEST MODE 6 W/ SOP NON-MODIFYING INSTS
                        ;*****************************************************************************
005106  005212          TST70:  INC     (R2)            ;UPDATE TEST NUMBER
005110  022712  000070          CMP     #70,(R2)        ;SEQUENCE ERROR?
005114  001021                  BNE     TST71-10        ;BR TO ERROR HALT ON SEQ ERROR
005116  005000                  CLR     R0              ;R0=0
005120  005010                  CLR     (R0)            ;LOC 0=0
005122  005110                  COM     (R0)            ;LOC 0=-1
005124  105100                  COMB    R0              ;R0=377
005126  000277                  SCC                     ;SET CC=0111
005130  000250                  CLN
005132  005760  177401          TST     -377(R0)        ;TRY TST W/ MODE 6
005136  102402                  BVS     SNM6A           ;CHECK CC=1000
005140  101401                  BLOS    SNM6A
005142  100404                  BMI     SNM6B
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 765           <====
005144                  SNM6A:
005144  012742  000134          MOV     #134,-(R2)      ;MOVE TO MAILBOX #  *******  134  *******
005150  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
005152  000000                  HALT                    ;CC'S INCORRECT
005154  105100          SNM6B:  COMB    R0              ;R0=0
005156  001404                  BEQ     TST71
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 757           <====
005160  012742  000135          MOV     #135,-(R2)      ;MOVE TO MAILBOX #  *******  135  *******
005164  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
005166  000000                  HALT                    ;TST MODE 6 INCORRECTLY CHANGED RC
                                                        ;  OR SEQUENCE ERROR
```

```
2518
2519                    ;*****************************************************************************
2520                    ;
2521                    ;       THIS TEST VERIFIES MODE 7 SOP NON-MODIFYING INSTRUCTIONS.
2522                    ;IT USES A POINTER TO LOC. 0 STORED AT LOC. 400 TO TST LOC. 0.
2523                    ;R0 IS SET TO 377 AND LOC. 0 IS TESTED THRU THE POINTER AT 400 USING
2524                    ;R0 AND AN OFFSET OF 1.
2525                    ;
2526                    ;*****************************************************************************
2527                    ;TEST 71        TEST MODE 7 W/ SOP NON-MODIFYING INSTS.
2528                    ;*****************************************************************************
2529  005170  005212    TST71:  INC     (R2)            ;UPDATE TEST NUMBER
2530  005172  022712    000071          CMP     #71,(R2)        ;SEQUENCE ERROR?
2531  005176  001021                    BNE     TST72-10        ;BR TO ERROR HALT ON SEQ ERROR
2532  005200  005000                    CLR     R0              ;R0=0
2533  005202  005010                    CLR     (R0)            ;LOC 0=0
2534  005204  005110                    COM     (R0)            ;LOC 0=-1
2535  005206  105100                    COMB    R0              ;R0=377
2536  005210  000277                    SCC                     ;CC=0111
2537  005212  000250                    CLN
2538  005214  005770    000001          TST     @1(R0)          ;TRY TST W/ MODE 7
2539  005220  102402                    BVS     SNM7A           ;CHECK CC=1000
2540  005222  101401                    BLOS    SNM7A
2541  005224  100404                    BMI     SNM7B
2542                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2543                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2544                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2545                                                    ;           WHICH FOLLOWS W/ 765           <====
2546  005226            SNM7A:
2547  005226  012742    000136          MOV     #136,-(R2)      ;MOVE TO MAILBOX #  *******  136  *******
2548  005232  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2549  005234  000000                    HALT                    ;CC'S NOT CORRECT
2550  005236  105100    SNM7B:  COMB    R0                      ;R0=0
2551  005240  001404                    BEQ     TST72
2552                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2553                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2554                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2555                                                    ;           WHICH FOLLOWS W/ 757           <====
2556  005242  012742    000137          MOV     #137,-(R2)      ;MOVE TO MAILBOX #  *******  137  *******
2557  005246  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2558  005250  000000                    HALT                    ;TST MODE 7 INCORRECTLY CHANGED RC
2559                                                    ;  OR SEQUENCE ERROR
```

```
2560
2561                    ;*****************************************************************************
2562                    ;
2563                    ;       THIS TEST VERIFIES MODE 0 DOUBLE OPERAND INSTRUCTIONS.  IT SETS
2564                    ;DATA IN R0 AND R4 AND USES THE ADD INSTRUCTION TO TEST THE DOP
2565                    ;MICROCODE.
2566                    ;
2567                    ;*****************************************************************************
2568                    ;TEST 72        TEST MODE 0 DOUBLE-OPERAND (DOP) INSTS.
2569                    ;*****************************************************************************
2570  005252  005212    ;ST72:  INC     (R2)            ;UPDATE TEST NUMBER
2571  005254  022712    000072          CMP     #72,(R2)        ;SEQUENCE ERROR?
2572  005260  001006                    BNE     TST73-10        ;BR TO ERROR HALT ON SEQ ERROR
2573  005262  005000                    CLR     R0              ;R0=0
2574  005264  005100                    COM     R0              ;R0=-1
2575  005266  005004                    CLR     R4              ;R4=0
2576  005270  060004                    ADD     R0,R4           ;TRY ADD: R4=-1
2577  005272  005204                    INC     R4              ;R4=0
2578  005274  001404                    BEQ     TST73
2579                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
2580                                                    ;           CONDITIONAL BRANCH INST. AND    <====
2581                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
2582                                                    ;           WHICH FOLLOWS W/ 772            <====
2583  005276  012742    000140          MOV     #140,-(R2)      ;MOVE TO MAILBOX #  *******  140  *******
2584  005302  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2585  005304  000000                    HALT                    ;ADD INST. FAILED W/ MODE 0
2586                                                    ;  OR SEQUENCE ERROR
2587
2588                    ;*****************************************************************************
2589                    ;
2590                    ;       THIS TEST VERIFIES THE MOVE INSTRUCTION WITH MODE 0 TO MODE 0.
2591                    ;THIS TEST IS NECESSARY BECAUSE THIS PARTICULAR INSTRUCTION UTILIZES UNIQUE
2592                    ;MICROCODE.
2593                    ;
2594                    ;*****************************************************************************
2595                    ;TEST 73        MOV MODE 0 TO MODE 0
2596                    ;*****************************************************************************
2597  005306  005212    ;ST73:  INC     (R2)            ;UPDATE TEST NUMBER
2598  005310  022712    000073          CMP     #73,(R2)        ;SEQUENCE ERROR?
2599  005314  001006                    BNE     TST74-10        ;BR TO ERROR HALT ON SEQ ERROR
2600  005316  005000                    CLR     R0              ;R0=0
2601  005320  005004                    CLR     R4              ;R4=0
2602  005322  005100                    COM     R0              ;R0=-1
2603  005324  010004                    MOV     R0,R4           ;TRY MOVE -1 TO R4
2604  005326  005204                    INC     R4              ;INC R4
2605  005330  001404                    BEQ     TST74
2606                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
2607                                                    ;           CONDITIONAL BRANCH INST. AND    \====
2608                                                    ;           REPLACE THE MOVE INSTRUCTION    \====
2609                                                    ;           WHICH FOLLOWS W/ 772            \====
2610  005332  012742    000141          MOV     #141,-(R2)      ;MOVE TO MAILBOX #  *******  141  *******
2611  005336  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2612  005340  000000                    HALT                    ;MOVE FAILED MODE 0 TO MODE 0
2613                                                    ;  OR SEQUENCE ERROR
```

```
2614
2615                    ;*****************************************************************************
2616                    ;
2617                    ;       THIS TEST QUICKLY VERIFIES THE REMAINING DOP MODIFYING INSTRUCTIONS
2618                    ;WITH MODE 0 TO PROVIDE A BASELINE FOR SUBSEQUENT TESTS.
2619                    ;SINGLE OPERAND INSTRUCTIONS ARE USED TO SET UP DATA IN R0 AND R4
2620                    ;BEFORE EACH OF THE SEVERAL DOP MODIFYING INSTRUCTIONS ARE USED AND
2621                    ;VERIFIED.
2622                    ;
2623                    ;*****************************************************************************
2624                    ;TEST 74        TEST ALL THE DOP INSTRUCTIONS W/ SOURCE MODE 0
2625                    ;*****************************************************************************
2626  005342  005212    ST74:   INC     (R2)            ;UPDATE TEST NUMBER
2627  005344  022712  000074    CMP     #74,(R2)        ;SEQUENCE ERROR?
2628  005350  001051            BNE     TST75-10        ;BR TO ERROR HALT ON SEQ ERROR
2629  005352  005000            CLR     R0              ;R0=0
2630  005354  010004            MOV     R0,R4           ;TRY MOVE MODE 0,0
2631  005356  001404            BEQ     DOP0A
2632                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2633                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2634                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2635                                                    ;           WHICH FOLLOWS W/ 775           <====
2636  005360  012742  000142          MOV     #142,-(R2)      ;MOVE TO MAILBOX #  *******  142  *******
2637  005364  005242            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2638  005366  000000            HALT                    ;Z-BIT NOT SET
2639  005370  005200    DOP0A:  INC     R0              ;R0=1
2640  005372  005100            COM     R0              ;R0=177776
2641  005374  005104            COM     R4              ;R4=177777
2642  005376  040004            BIC     R0,R4           ;TRY BIC: R4=1
2643  005400  005304            DEC     R4              ;R4=0
2644  005402  001404            BEQ     DOP0B
2645                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2646                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2647                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2648                                                    ;           WHICH FOLLOWS W/ 763           <====
2649  005404  012742  000143          MOV     #143,-(R2)      ;MOVE TO MAILBOX #  *******  143  *******
2650  005410  005242            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2651  005412  000000            HALT                    ;BIC CLEAR RESULT INCORRECT
2652  005414  050004    DOP0B:  BIS     R0,R4           ;TRY BIS: R4=177777
2653  005416  005204            INC     R4
2654  005420  005204            INC     R4              ;R4=0
2655  005422  001404            BEQ     DOP0C
2656                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2657                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2658                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2659                                                    ;           WHICH FOLLOWS W/ 753           <====
2660  005424  012742  000144          MOV     #144,-(R2)      ;MOVE TO MAILBOX #  *******  144  *******
2661  005430  005242            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2662  005432  000000            HALT                    ;RESULT OF BIS INCORRECT
2663  005434  005000    DOP0C:  CLR     R0              ;R0=0
2664  005436  105100            COMB    R0              ;R0=377
2665  005440  005004            CLR     R4              ;R4=0
2666  005442  005104            COM     R4              ;R4=177777
2667  005444  040004            BIC     R0,R4           ;R4=177400
2668  005446  060004            ADD     R0,R4           ;TRY ADD: R4=177777
2669  005450  005204            INC     R4              ;R4=0
```

```
 2670  005452  001404                    BEQ     DOPOD
 2671                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    =
 2672                                                    ;           CONDITIONAL BRANCH INST. AND    =
 2673                                                    ;           REPLACE THE MOVE INSTRUCTION    =
 2674                                                    ;           WHICH FOLLOWS W/ 737            ====
 2675  005454  012742  000145                MOV     #145,-(R2)  ;MOVE TO MAILBOX #  *******  145  *******
 2676  005460  005242                    INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
 2677  005462  000000                    HALT                ;RESULT OF ADD INCORRECT
 2678  005464  160004            DOPOD:  SUB     R0,R4       ;177401=R4
 2679  005466  105404                    NEGB    R4          ;R4=177777
 2680  005470  005204                    INC     R4          ;R0=0
 2681  005472  001404                    BEQ     TST75
 2682                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
 2683                                                    ;           CONDITIONAL BRANCH INST. AND    <====
 2684                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
 2685                                                    ;           WHICH FOLLOWS W/ 727            <====
 2686  005474  012742  000146                MOV     #146,-(R2)  ;MOVE TO MAILBOX #  *******  146  *******
 2687  005500  005242                    INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
 2688  005502  000000                    HALT                ;RESULT OF SUB INCORRECT
 2689                                                    ;  OR SEQUENCE ERROR
```

```
2690
2691                    ;*****************************************************************************
2692                    ;
2693                    ;       THIS TEST VERIFIES MODE 0 DOP NON-MODIFYING INSTRUCTIONS.
2694                    ;R0 AND R4 ARE PRESET TO 0 AND 1 RESPECTIVELY.  COMPARE INSTRUCTIONS ARE
2695                    ;THEN EXECUTED AND CHECKED.  FIRST R4 IS COMPARED TO R0 THEN R0 TO R4.
2696                    ;
2697                    ;*****************************************************************************
2698                    ;TEST 75        TEST DOP NON-MODIFYING INST. W/ SOURCE MODE 0
2699                    ;*****************************************************************************
2700  005504  005212    TST75:  INC     (R2)            ;UPDATE TEST NUMBER
2701  005506  022712    000075          CMP     #75,(R2)        ;SEQUENCE ERROR?
2702  005512  001042                    BNE     TST76-10        ;BR TO ERROR HALT ON SEQ ERROR
2703  005514  005000                    CLR     R0              ;R0=0
2704  005516  005004                    CLR     R4              ;R4=0
2705  005520  005204                    INC     R4              ;R4=1
2706  005522  020400                    CMP     R4,R0           ;TRY COMPARE R4 TO R0
2707  005524  003004                    BGT     DNM1
2708                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2709                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2710                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2711                                                    ;           WHICH FOLLOWS W/ 773           <====
2712  005526  012742    000147          MOV     #147,-(R2)      ;MOVE TO MAILBOX #  *******  147  *******
2713  005532  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2714  005534  000000                    HALT                    ;CC'S NOT CORRECT FOR CMP
2715  005536  020004    DNM1:   CMP     R0,R4           ;TRY COMPARE R0 TO R4
2716  005540  002404                    BLT     DNM2
2717                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2718                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2719                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2720                                                    ;           WHICH FOLLOWS W/ 765           <====
2721  005542  012742    000150          MOV     #150,-(R2)      ;MOVE TO MAILBOX #  *******  150  *******
2722  005546  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2723  005550  000000                    HALT                    ;CC'S NOT CORRECT FOR CMP
2724  005552  005200    DNM2:   INC     R0              ;R4=0
2725  005554  020400                    CMP     R4,R0           ;TRY COMPARE R4=1 TO R0=1
2726  005556  001404                    BEQ     DNM3
2727                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2728                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2729                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2730                                                    ;           WHICH FOLLOWS W/ 756           <====
2731  005560  012742    000151          MOV     #151,-(R2)      ;MOVE TO MAILBOX #  *******  151  *******
2732  005564  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2733  005566  000000                    HALT                    ;CC'S NOT CORRECT (Z=1) FOR CMP
2734  005570  005000    DNM3:   CLR     R0              ;R0=0
2735  005572  005100                    COM     R0              ;R0=177777
2736  005574  005004                    CLR     R4              ;R4=0
2737  005576  030004                    BIT     R0,R4           ;TRY BIT R0 TO R4
2738  005600  001404                    BEQ     DNM4
2739                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2740                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2741                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2742                                                    ;           WHICH FOLLOWS W/ 745           <====
2743  005602  012742    000152          MOV     #152,-(R2)      ;MOVE TO MAILBOX #  *******  152  *******
2744  005606  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2745  005610  000000                    HALT                    ;CC'S NOT CORRECT FOR BIT
```

```
2746  005612  005304                  DN74:   DEC     R4              ;R4=177777
2747  005614  030004                          BIT     R0,R4           ;TRY BIT AGAIN
2748  005616  100404                          BMI     TST76
2749                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
2750                                                                  ;           CONDITIONAL BRANCH INST. AND    <====
2751                                                                  ;           REPLACE THE MOVE INSTRUCTION    <====
2752                                                                  ;           WHICH FOLLOWS W/ 736            <====
2753  005620  012742  000153                      MOV     #153,-(R2)  ;MOVE TO MAILBOX #  *******  153  *******
2754  005624  005242                              INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
2755  005626  000000                              HALT                ;CC'S NOT CORRECT FOR BIT
2756                                                                  ;  OR SEQUENCE ERROR
2757
2758                                  ;*****************************************************************************
2759                                  ;
2760                                  ;       THIS TEST VERIFIES MODE 1 DOP INSTRUCTIONS.  R0 IS SET TO -1
2761                                  ;AND LOC 0 TO 1.  R4 IS THEN CLEARED AND USED TO POINT TO LOC 0.
2762                                  ;IN THE ADD MODE 1 INSTRUCTION, LOC 0 IS ADDED TO R0 AND THE
2763                                  ;RESULTS VERIFIED.
2764                                  ;
2765                                  ;*****************************************************************************
2765                                  ;TEST 76        TEST MODE 1 W/ DOP INST.
2767                                  ;*****************************************************************************
2768  005630  005212                  TST76:  INC     (R2)            ;UPDATE TEST NUMBER
2769  005632  022712  000076                  CMP     #76,(R2)        ;SEQUENCE ERROR?
2770  005636  001007                          BNE     TST77-10        ;BR TO ERROR HALT ON SEQ ERROR
2771  005640  005000                          CLR     R0              ;R0=0
2772  005642  005100                          COM     R0              ;R0=177777
2773  005644  005004                          CLR     R4              ;R4=0
2774  005646  005014                          CLR     (R4)            ;LOC 0=0
2775  005650  005214                          INC     (R4)            ;LOC 0=1
2776  005652  061400                          ADD     (R4),R0         ;TRY ADD SOURCE MODE 1
2777  005654  001404                          BEQ     TST77
2778                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
2779                                                                  ;           CONDITIONAL BRANCH INST. AND    <====
2780                                                                  ;           REPLACE THE MOVE INSTRUCTION    <====
2781                                                                  ;           WHICH FOLLOWS W/ 771            <====
2782  005656  012742  000154                      MOV     #154,-(R2)  ;MOVE TO MAILBOX #  *******  154  *******
2783  005662  005242                              INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
2784  005664  000000                              HALT                ;RESULT OF ADD INCORRECT
2785                                                                  ;  OR SEQUENCE ERROR
```

```
2786
2787
2788
2789
2790
2791
2792
2793
2794
2795
2796  005666  005212                  TST77:  INC     (R2)            ;UPDATE TEST NUMBER
2797  005670  022712  000077                  CMP     #77,(R2)        ;SEQUENCE ERROR?
2798  005674  001007                        BNE     TST100-10       ;BR TO ERROR HALT ON SEQ ERROR
2799  005676  005000                        CLR     R0              ;R0=0
2800  005700  005010                        CLR     (R0)            ;LOC. 0=0
2801  005702  005110                        COM     (R0)            ;LOC. 0=177777
2802  005704  005004                        CLR     R4              ;R4=0
2803  005706  151004                        BISB    (R0),R4         ;TRY MODE 1- EVEN BYTE W/ DOP
2804  005710  105104                        COMB    R4              ;R4=0
2805  005712  001404                        BEQ     TST100
2806                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2807                                                                ;           CONDITIONAL BRANCH INST. AND   <====
2808                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
2809                                                                ;           WHICH FOLLOWS W/ 771           <====
2810  005714  012742  000155                  MOV     #155,-(R2)      ;MOVE TO MAILBOX #  *******  155  *******
2811  005720  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2812  005722  000000                        HALT                    ;RESULT OF BISB IS INCORRECT
2813                                                                ;  OR SEQUENCE ERROR
```

```
2814
2815                            ;*****************************************************************************
2816                            ;
2817                            ;       THIS TEST VERIFIES MODE 1 DOP NON-MODIFYING INSTRUCTIONS
2818                            ;WHICH ADDRESS EVEN BYTES.  LOC. 0 IS SET TO -1 AND R0 IS CLEARED
2819                            ;AND USED AS THE ADDRESSING REGISTER.  R4 IS SET TO 377 AND A
2820                            ;MODE 1,0 CMPB INSTRUCTION IS USED THE RESULTS VERIFIED.
2821                            ;
2822                            ;*****************************************************************************
2823                            ;TEST 100       TEST MODE 1 - EVEN BYTE W/ DOP NON-MODIFYING INST.
2824                            ;*****************************************************************************
2825  005724  005212            TST100: INC     (R2)            ;UPDATE TEST NUMBER
2826  005726  022712  000100           CMP     #100,(R2)       ;SEQUENCE ERROR?
2827  005732  001007                   BNE     TST101-10       ;BR TO ERROR HALT ON SEQ ERROR
2828  005734  005000                   CLR     R0              ;R0=0
2829  005736  005010                   CLR     (R0)            ;LOC 0=0
2830  005740  005110                   COM     (R0)            ;LOC 0=177777
2831  005742  005004                   CLR     R4              ;R4=0
2832  005744  105104                   COMB    R4              ;R4=377
2833  005746  121004                   CMPB    (R0),R4         ;TRY MODE 1 - EVEN BYTE W/ DOP NON-MODIFYING
2834  005750  001404                   BEQ     TST101
2835                                                           ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2836                                                           ;           CONDITIONAL BRANCH INST. AND   <====
2837                                                           ;           REPLACE THE MOVE INSTRUCTION   <====
2838                                                           ;           WHICH FOLLOWS W/ 771           <====
2839  005752  012742  000156           MOV     #156,-(R2)      ;MOVE TO MAILBOX #  *******  156  *******
2840  005756  005242                   INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2841  005760  000000                   HALT                    ;RESULT OF CMPB INCORRECT
2842                                                           ;  OR SEQUENCE ERROR
```

```
2843
2844                            ;*****************************************************************************
2845                            ;
2846                            ;       THIS TEST VERIFIES MODE 1,0 MOVB INSTRUCTIONS
2847                            ;WHICH ADDRESS EVEN BYTES.  LOC. 0 IS SET TO 177400, R0 IS CLEARED AND
2848                            ;R4 IS SET TO -1.  MOVB ARE USED TO MOVE BYTE 0 TO R4.  THIS
2849                            ;VERIFIES THAT THE PROPER BYTE WAS SELECTED AND THAT THE SIGN-X-TEND
2850                            ;FUNCTION WITH MODE 0.
2851                            ;       THEN LOC. 0 IS COMPLEMENTED AND THE SAME PROCEDURE EXERCISES
2852                            ;THE LOGIC FOR COMPLEMENTARY DATA.
2853                            ;       THIS TEST EXERCISES UNIQUE MICROCODE.
2854                            ;
2855                            ;*****************************************************************************
2856                            ;TEST 101       TEST MOV INSTRUCTION MODE 1,0 EVEN BYTE
2857                            ;*****************************************************************************
2858  005762  005212            TST101: INC     (R2)            ;UPDATE TEST NUMBER
2859  005764  022712  000101            CMP     #101,(R2)       ;SEQUENCE ERROR?
2860  005770  001020                    BNE     TST102-10       ;BR TO ERROR HALT ON SEQ ERROR
2861  005772  005000                    CLR     R0              ;R0=0
2862  005774  005010                    CLR     (R0)            ;LOC 0=0
2863  005776  105110                    COMB    (R0)            ;LOC 0=177400
2864  006000  005110                    COM     (R0)
2865  006002  005004                    CLR     R4              ;R4=0
2866  006004  005104                    COM     R4              ;R4=177777
2867  006006  111004                    MOVB    (R0),R4         ;R4=0
2868  006010  005704                    TST     R4              ;CHECK SIGN OF WORD
2869  006012  001404                    BEQ     DOP1
2870                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2871                                                            ;           CONDITIONAL BRANCH INST. AND   <====
2872                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
2873                                                            ;           WHICH FOLLOWS W/ 767           <====
2874  006014  012742  000157                  MOV     #157,-(R2)      ;MOVE TO MAILBOX #  *******  157  *******
2875  006020  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2876  006022  000000                    HALT                    ;MOVB SHOULD SIGN X-TEND
2877  006024  005110            DOP1:   COM     (R0)            ;LOC 0=177777
2878  006026  111004                    MOVB    (R0),R4         ;DO MOVB W/ EVEN BYTE
2879  006030  100404                    BMI     TST102
2880                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2881                                                            ;           CONDITIONAL BRANCH INST. AND   <====
2882                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
2883                                                            ;           WHICH FOLLOWS W/ 760           <====
2884  006032  012742  000160                  MOV     #160,-(R2)      ;MOVE TO MAILBOX #  *******  160  *******
2885  006036  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2886  006040  000000                    HALT                    ;MOVB SHOULD SIGN X-TEND
2887                                                            ;  OR SEQUENCE ERROR
```

```
                        ;*****************************************************************************
                        ;
                        ;       THIS TEST VERIFIES MODE 1 DOP INSTRUCTIONS WHICH REFERENCE
                        ;ODD BYTES.  LOC. 0 IS SET TO 177400.  R0 IS SET TO 0 AND R4 IS
                        ;SET TO 1.  THE BISB INSTRUCTION USES THE DATA IN BYTE 1 TO SET BYTE 0.
                        ;THE RESULT IS CHECKED BY INCREMENTING THE WORD (LOC. 0) TO ZERO.
                        ;
                        ;*****************************************************************************
                        ;TEST 102       TEST MODE 1-ODD BYTE W/ DOP INSTS.
                        ;*****************************************************************************
006042  005212          TST102: INC     (R2)            ;UPDATE TEST NUMBER
006044  022712  000102          CMP     #102,(R2)       ;SEQUENCE ERROR?
006050  001010                  BNE     TST103-10       ;BR TO ERROR HALT ON SEQ ERROR
006052  005000                  CLR     R0              ;R0=0
006054  005010                  CLR     (R0)            ;LOC. 0=0
006056  005004                  CLR     R4              ;R4=0
006060  005204                  INC     R4              ;R4=1
006062  105114                  COMB    (R4)            ;LOC. 0=177400
006064  151410                  BISB    (R4),(R0)       ;TRY TO BIS LOW ORDER BITS W/ MODE 1
006066  005210                  INC     (R0)            ;CHECK RESULT
006070  001404                  BEQ     TST103
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <== =
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W' 770           <====
006072  012742  000161          MOV     #161,-(R2)      ;MOVE TO MAILBOX #  *******  161  *******
006076  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006100  000000                  HALT                    ;RESULT OF BISB INCORRECT
                                                        ;  OR SEQUENCE ERROR
```

```
2918
2919                                    ;*****************************************************************************
2920                                    ;
2921                                    ;       THIS TEST VERIFIES MODE 2 DOP INSTRUCTIONS.  LOC. 0 IS SET TO -1.
2922                                    ;R0 IS CLEARED AND USED AS THE MODE 2 ADDRESSING REGISTER TO MOVE LOC. 0
2923                                    ;TO R7.  THE DATA RESULTS ARE VERIFIED AND THE INCREMENTING OF THE REGISTER
2924                                    ;IS CHECKED.
2925                                    ;
2926                                    ;*****************************************************************************
2927                                    ;TEST 103       TEST MODE 2 W/ DOP INSTS.
2928                                    ;*****************************************************************************
2929  006102  005212                    TST103: INC     (R2)            ;UPDATE TEST NUMBER
2930  006104  022712  000103                    CMP     #103,(R2)       ;SEQUENCE ERROR?
2931  006110  001015                            BNE     TST104-10       ;BR TO ERROR HALT ON SEQ ERROR
2932  006112  005000                            CLR     R0              ;R0=0
2933  006114  005010                            CLR     (R0)            ;LOC. 0=0
2934  006116  005110                            COM     (R0)            ;LOC. 0=177777
2935  006120  012004                            MOV     (R0)+,R4        ;TRY MOVE MODE 2,0
2936  006122  005204                            INC     R4              ;CHECK R4
2937  006124  001404                            BEQ     DOP2
2938                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2939                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2940                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2941                                                                    ;           WHICH FOLLOWS W/ 772           <====
2942  006126  012742  000162                      MOV     #162,-(R2)    ;MOVE TO MAILBOX #  *******  162  *******
2943  006132  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2944  006134  000000                            HALT                    ;RESULT OF MOV INST INCORRECT
2945  006136  005300                    DOP2:   DEC     R0              ;TEST R0 AFTER MODE 2
2946  006140  005300                            DEC     R0
2947  006142  001404                            BEQ     TST104
2948                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
2949                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
2950                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
2951                                                                    ;           WHICH FOLLOWS W/ 763           <====
2952  006144  012742  000163                      MOV     #163,-(R2)    ;MOVE TO MAILBOX #  *******  163  *******
2953  006150  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
2954  006152  000000                            HALT                    ;REGISTER NOT INCREMENTED IN MODE 2
2955                                                                    ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 2 DOP BYTE INSTRUCTIONS WHICH ADDRESS
                                ;EVEN BYTES.  LOC. 0 IS SET TO -1.  R0 IS CLEARED AND USED AS THE
                                ;ADDRESSING REGISTER IN A TEST WHICH TRIES TO CLEAR BYTE 1 USING
                                ;BYTE 0 DATA AND A BICB.  UNIQUE IN THIS TEST IS USE OF THE
                                ;SAME ADDRESSING REGISTER FOR BOTH SOURCE AND DESTINATION.  THE SOURCE AND
                                ;DESTINATION IS CHECKED TO INSURE PROPER FUNCTIONING.
                                ;
                                ;*****************************************************************************
                                ;TEST 104       TEST MODE 2 - EVEN BYTE W/ DOP INST.
                                ;*****************************************************************************
006154  005212          TST104: INC     (R2)            ;UPDATE TEST NUMBER
006156  022712  000104          CMP     #104,(R2)       ;SEQUENCE ERROR?
006162  001016                  BNE     TST105-10       ;BR TO ERROR HALT ON SEQ ERROR
006164  005000                  CLR     R0              ;R0=0
006166  010010                  MOV     R0,(R0)         ;LOC. 0=0
006170  005110                  COM     (R0)            ;LOC. 0=177777
006172  142010                  BICB    (R0)+,(R0)      ;TRY TO CLEAR BYTE 1 FROM BYTE 0 W/ BICB
006174  105737  000001          TSTB    @#1             ;CHECK RESULT
006200  001404                  BEQ     DOPB2A
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 771           <====
006202  012742  000164                  MOV     #164,-(R2)      ;MOVE TO MAILBOX #  *******  164  *******
006206  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006210  000000                  HALT                    ;BICB DESTINATION INCORRECT
006212  105137  000000  DOPB2A: COMB    @#0             ;CHECK BICB SOURCE
006216  001404                  BEQ     TST105
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 762           <====
006220  012742  000165          MOV     #165,-(R2)      ;MOVE TO MAILBOX #  *******  165  *******
006224  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006226  000000                  HALT                    ;BICB SOURCE INCORRECTLY CHANGED
                                                        ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 2 DOP BYTE INSTRUCTIONS WHICH REFERENCE
                                ;ODD BYTES.  R0 IS SET TO 1, LOC. 0 IS SET TO 177400, AND R4 IS CLEARED.
                                ;A MODE 2 MOVB USES R0 TO MOVE BYTE 1 TO R4.  AN INCREMENT
                                ;IS USED TO CHECK THAT THE PROPER BYTE WAS MOVED AND SIGN X-TENDED.
                                ;
                                ;*****************************************************************************
                                ;TEST 105       TEST MODE 2 - ODD BYTE W/ DOP INST.
                                ;*****************************************************************************
006230  005212        000105  ;ST105: INC     (R2)            ;UPDATE TEST NUMBER
006232  022712                        CMP     #105,(R2)       ;SEQUENCE ERROR?
006236  001017                        BNE     TST106-10       ;BR TO ERROR HALT ON SEQ ERROR
006240  005000                        CLR     R0              ;R0=0
006242  005004                        CLR     R4              ;R4=0
006244  005010                        CLR     (R0)            ;LOC. 0=0
006246  005110                        COM     (R0)            ;LOC. 0=177777
006250  105120                        COMB    (R0)+           ;LOC 0=177400; R0=1
006252  112004                        MOVB    (R0)+,R4        ;TRY DOP MODE 2 W/ ODD BYTE
006254  005204                        INC     R4              ;CHECK RESULT OF MOVB
006256  001404                        BEQ     DOPB2B
                                                            ;  TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;            CONDITIONAL BRANCH INST. AND   <====
                                                            ;            REPLACE THE MOVE INSTRUCTION   <====
                                                            ;            WHICH FOLLOWS W/ 770           <====
006260  012742  000166                  MOV     #166,-(R2)      ;MOVE TO MAILBOX #  *******  166  *******
006264  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006266  000000                          HALT                    ;RESULT OF MOVB INCORRECT
006270  005740                DOPB2B: TST     -(R0)           ;BUMP R0 DOWN BY 2
006272  005700                        TST     R0              ;CHECK R0
006274  001404                        BEQ     TST106
                                                            ;  TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;            CONDITIONAL BRANCH INST. AND   <====
                                                            ;            REPLACE THE MOVE INSTRUCTION   <====
                                                            ;            WHICH FOLLOWS W/ 761           <====
006276  012742  000167                  MOV     #167,-(R2)      ;MOVE TO MAILBOX #  *******  167  *******
006302  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006304  000000                          HALT                    ;MODE 2 BYTE DID NOT INCREMENT REG. CORRECTLY
                                                            ;  OR SEQUENCE ERROR
```

```
                                        ;*****************************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES MODE 3 DOUBLE-OPERAND INSTRUCTIONS.
                                        ;LOC. 0 IS LOADED WITH ALTERNATING ZEROES AND ONES; AND R0 IS LOADED
                                        ;WITH ALTERNATING ONES AND ZEROES.  A MODE 3 BIS IS USED TO SET R0
                                        ;TO -1 BY USING LOC. 0 AS THE SOURCE TO BIS THE ZEROES IN R0.  THE
                                        ;RESULT IS TESTED BY INCREMENTING R0 AND CHECKING FOR ZERO.
                                        ;
                                        ;*****************************************************************************
                                        ;TEST 106       TEST MODE 3 W/ DOP INSTS.
                                        ;*****************************************************************************
006306  005212                  TST106: INC     (R2)            ;UPDATE TEST NUMBER
006310  022712  000106                  CMP     #106,(R2)       ;SEQUENCE ERROR?
006314  001011                          BNE     TST107-10       ;BR TO ERROR HALT ON SEQ ERROR
006316  012737  052525  000000          MOV     #052525,@#0     ;MOVE 52525 TO LOC. 0
006324  012700  125252                  MOV     #125252,R0      ;SET ALT. ONE AND ZERO IN R0
006330  053700  000000                  BIS     @#0,R0          ;TRY TO SET ALL OTHER BITS W/ MODE 3
006334  005200                          INC     R0              ;TEST RESULT
006336  001404                          BEQ     TST107
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 767           <====
006340  012742  000170                  MOV     #170,-(R2)      ;MOVE TO MAILBOX #  *******  170  *******
006344  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006346  000000                          HALT                    ;BIS W/ MODE 3 INCORRECT RESULT
                                                                ;  OR SEQUENCE ERROR
                                        ;*****************************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES MODE 3 DOUBLE OPERAND BYTE INSTRUCTIONS WHICH
                                        ;ADDRESS EVEN BYTES.  BYTE 0 IS SET TO ALTERNATING 1'S AND 0'S; BYTE 1,
                                        ;ALTERNATING 0'S AND 1'S.  R0 IS CLEARED AND A BISB IS USED TO
                                        ;SET THE LOW BYTE OF R0 TO 252.
                                        ;
                                        ;*****************************************************************************
                                        ;TEST 107       TEST MODE 3 - EVEN BYTE W/ DOP INSTS.
                                        ;*****************************************************************************
006350  005212                  TST107: INC     (R2)            ;UPDATE TEST NUMBER
006352  022712  000107                  CMP     #107,(R2)       ;SEQUENCE ERROR?
006356  001011                          BNE     TST110-10       ;BR TO ERROR HALT ON SEQ ERROR
006360  012737  052652  000000          MOV     #52652,@#0      ;MOVE 1'S AND 0' PATTERN TO LOC. 0
006366  005000                          CLR     R0              ;R0=0
006370  153700  000000                  BISB    @#0,R0          ;TRY R0=252 W/ MODE 3 - EVEN BYTE
006374  022700  000252                  CMP     #252,R0         ;BISB W/ EVEN BYTE SUCCESSFUL?
006400  001404                          BEQ     TST110
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 767           <====
006402  012742  000171                  MOV     #171,-(R2)      ;MOVE TO MAILBOX #  *******  171  *******
006406  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006410  000000                          HALT                    ;BISB W/ MODE 3 - EVEN BYTE FAILED
                                                                ;  OR SEQUENCE ERROR
```

```
                        ;*****************************************************************************
                        ;
                        ;       THIS TEST VERIFIES MODE 3 DOUBLE OPERAND BYTE INSTRUCTIONS
                        ;WHICH ADDRESS ODD BYTES.  THE SAME PROCEDURE USED IN PREVIOUS
                        ;TEST IS USED HERE.  THIS TIME BYTE 1 IS USED AS THE SOURCE BYTE.
                        ;THE EXPECTED RESULT IS:  R0 = 125.
                        ;
                        ;*****************************************************************************
                        ;TEST 110       TEST MODE 3 - ODD BYTE W/ DOP INSTS.
                        ;*****************************************************************************
006412  005212          TST110: INC     (R2)            ;UPDATE TEST NUMBER
006414  022712  000110          CMP     #110,(R2)       ;SEQUENCE ERROR?
006420  001011                  BNE     TST111-10       ;BR TO ERROR HALT ON SEQ ERROR
006422  012737  052652  000000  MOV     #52652,@#0      ;MOVE 1'S AND 0'S PATTERN TO LOC 0
006430  005000                  CLR     R0              ;R0=0
006432  153700  000001          BISB    @#1,R0          ;TRY R0=152 W/ MODE 3 - ODD BYTE
006436  022700  000125          CMP     #125,R0         ;R0=125?
006442  001404                  BEQ     TST111
                                                        ; TO SCOPE. CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 767           <====
006444  012742  000172          MOV     #172,-(R2)      ;MOVE TO MAILBOX #  *******  172  *******
006450  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006452  000000                  HALT                    ;BISB W/ MODE 3 - ODD BYTE FAILED
                                                        ;  OR SEQUENCE ERROR
                        ;*****************************************************************************
                        ;
                        ;       THIS TEST VERIFIES MODE 4 DOUBLE OPERAND INSTRUCTIONS.
                        ;THE TEST USES MODE 4 ADDRESSING WITH REGISTER 0 TO MOVE THRU A
                        ;TABLE OF OPERANDS.  THE TABLE OF OPERANDS AND THE WORK LOCATION IS
                        ;STORED FOLLOWING THE TEST CODE.  A SERIES OF 5 DOP INSTRUCTIONS UTILIZES
                        ;THE DATA IN THE TABLE TO CYCLE THE WORK LOCATION THRU A SET OF
                        ;VALUE.  THE DATA HAS BEEN CHOSEN TO INSURE THAT NO SINGLE ERROR WILL
                        ;GO UNDETECTED.  WORD AND BYTE INSTRUCTION ACCESSING BOTH EVEN AND
                        ;ODD ADDRESSES ARE USED IN THE TEST.  THE LISTING SHOWS THE
                        ;EXPECTED INTERMEDIATE RESULT AS EACH INSTRUCTION IS EXECUTED.
                        ;
                        ;*****************************************************************************
                        ;TEST 111       TEST MODE 4 W/ DOP INSTS.
                        ;*****************************************************************************
006454  005212          TST111: INC     (R2)            ;UPDATE TEST NUMBER
006456  022712  000111          CMP     #111,(R2)       ;SEQUENCE ERROR?
006462  001015                  BNE     DOP4            ;BR TO ERROR HALT ON SEQ ERROR
006464  012700  006536          MOV     #TBL1,R0        ;INITIALIZE R0
006470  014037  006536          MOV     -(R0),@#TBL1    ;TBL1=125252
006474  064037  006536          ADD     -(R0),@#TBL1    ;TBL1=000377
006500  144037  006536          BICB    -(R0),@#TBL1    ;TBL1=000252
006504  154037  006537          BISB    -(R0),@#TBL1+1  ;TBL1=125252
006510  024037  006536          CMP     -(R0),@#TBL1    ;CHECK RESULT
006514  001411                  BEQ     TST112
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION    ====
```

```
3145                                                    ;           WHICH FOLLOWS W/ 763
3146  006516                                    DOP4:
3147  006516  012742  000173                          MOV     #173,-(R2)      ;MOVE TO MAILBOX #  *******  173  *******
3148  006522  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3149  006524  000000                                  HALT                    ;RESULT OF MODE 4 INSTS. INCORPECT
3150                                                                          ;  OR SEQUENCE ERROR
3151
3152  006526  125252                                  125252
3153  006530  052652                                  52652
3154  006532  053125                                  53125
3155  006534  125252                                  125252
3156  006536  000000                    TBL1:         0
3157
3158                                    ;*****************************************************************************
3159                                    ;
3160                                    ;       THIS TEST VERIFIES MODE 5 DOUBLE OPERAND INSTRUCTIONS.
3161                                    ;THE TEST USES AN ADDRESS TABLE STORED FOLLOWING THE TEST CODE.
3162                                    ;THIS TABLE IS SIMPLY A TABLE OF ADDRESS POINTERS WHICH ADDRESS
3163                                    ;THE DATA TABLE USED IN THE PREVIOUS TEST.  THE TEST IS IDENTICAL TO
3164                                    ;THE PREVIOUS TEST EXCEPT THE DATA IS REFERENCED USING THIS ADDRESS
3165                                    ;TABLE AND MODE 5 ADDRESSING.  (SEE PREVIOUS TEST).
3166                                    ;
3167                                    ;*****************************************************************************
3168                                    ;TEST 112       TEST MODE 5 W/ DOP INSTS.
3169                                    ;*****************************************************************************
3170  006540  005212                    TST112: INC     (R2)            ;UPDATE TEST NUMBER
3171  006542  022712  000112                    CMP     #112,(R2)       ;SEQUENCE ERROR?
3172  006546  001015                            BNE     DOP5            ;BR TO ERROR HALT ON SEQ ERROR
3173  006550  012700  006624                    MOV     #TBL2+2,R0      ;INITIALIZE R0
3174  006554  015037  006536                    MOV     @-(R0),@#TBL1   ;TBL1=125252
3175  006560  065037  006536                    ADD     @-(R0),@#TBL1   ;TBL1=000377
3176  006564  145037  006536                    BICB    @-(R0),@#TBL1   ;TBL1=000252
3177  006570  155037  006537                    BISB    @-(R0),@#TBL1+1 ;TBL1=125252
3178  006574  025037  006536                    CMP     @-(R0),@#TBL1   ;CHECK RESULT
3179  006600  001411                            BEQ     TST113
3180                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3181                                                                    ;           CONDITIONAL BRANCH INST. AND   <====
3182                                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
3183                                                                    ;           WHICH FOLLOWS W/ 763           <====
3184  006602                                    DOP5:
3185  006602  012742  000174                          MOV     #174,-(R2)      ;MOVE TO MAILBOX #  *******  174  *******
3186  006606  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3187  006610  000000                                  HALT                    ;RESULT OF MODE 5 INSTS. INCORRECT
3188                                                                          ;  OR SEQUENCE ERROR
3189  006612  006526                                  TBL1-10
3190  006614  006530                                  TBL1-6
3191  006616  006531                                  TBL1-5
3192  006620  006532                                  TBL1-4
3193  006622  006534                    TBL2:         TBL1-2
```

```
                                        ;*****************************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES MODE 6 DOUBLE OPERAND INSTRUCTIONS.
                                        ;IT USES THE SAME DATA AS THAT USED IN THE MODE 4 TESTS.
                                        ;THIS TIME THE DATA IS ACCESSED USING MODE 6.  R0 IS SET
                                        ;TO POINT TO THE MIDDLE OF THE TABLE.  THE TABLE IS ACCESSED FROM
                                        ;BOTTOM TO TOP BY VARYING THE OFFSET IN THE MODE 6 INSTRUCTIONS.
                                        ;THE DATA RESULTS ARE IDENTICAL TO THOSE EXPECTED IN THE MODE 4
                                        ;TESTS.
                                        ;
                                        ;*****************************************************************************
                                        ;TEST 113       TEST MODE 6 W/ DOP INSTS.
                                        ;*****************************************************************************
006624  005212                  TST113: INC     (R2)            ;UPDATE TEST NUMBER
006626  022712  000113                  CMP     #113,(R2)       ;SEQUENCE ERROR?
006632  001022                          BNE     TST114-10       ;BR TO ERROR HALT ON SEQ ERROR
006634  012700  006532                  MOV     #TBL1-4,R0      ;INITIALIZE R0
006640  016037  000002  006536          MOV     2(R0),@#TBL1    ;TBL1=125252
006646  066037  000000  006536          ADD     0(R0),@#TBL1    ;TBL1=000377
006654  146037  177777  006536          BICB    -1(R0),@#TBL1   ;TBL1=000252
006662  156037  177776  006537          BISB    -2(R0),@#TBL1+1 ;TBL1=125252
006670  026037  177774  006536          CMP     -4(R0),@#TBL1   ;CHECK RESULT
006676  001404                          BEQ     TST114
                                        ;  TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                        ;           WHICH FOLLOWS W/ 756           <====
006700  012742  000175                  MOV     #175,-(R2)      ;MOVE TO MAILBOX #  *******  175  *******
006704  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
006706  000000                          HALT                    ;RESULT OF MODE 6 INSTS. INCORRECT
                                        ;  OR SEQUENCE ERROR
;*****************************************************************************
;
;       THIS TEST VERIFIES MODE 7 DOUBLE OPERAND INSTRUCTIONS.
;THIS TEST USES THE SAME ADDRESS TABLE AND DATA TABLE USED BY
;THE MODE 5 TESTS.  THIS TIME THE DATA IS ACCESSED USING MODE 7.
;R0 IS SET TO POINT TO THE MIDDLE OF THE ADDRESS TABLE IN THE MODE 5
;TEST.  THE TABLE IS ACCESSED FROM BOTTOM TO TOP BY VARYING THE OFFSET
;IN THE MODE 7 INSTRUCTIONS.  THE DATA RESULTS ARE IDENTICAL TO
;THOSE EXPECTED IN THE MODE 5 TESTS.
;
;*****************************************************************************
;TEST 114       TEST MODE 7 W/ DOP INSTS.
;*****************************************************************************
006710  005212                  TST114: INC     (R2)            ;UPDATE TEST NUMBER
006712  022712  000114                  CMP     #114,(R2)       ;SEQUENCE ERROR?
006716  001022                          BNE     TST115-10       ;BR TO ERROR HALT ON SEQ ERROR
006720  012700  006616                  MOV     #TBL2-4,R0      ;INITIALIZE R0
006724  017037  000004  006536          MOV     @4(R0),@#TBL1   ;TBL1=125252
006732  067037  000002  006536          ADD     @2(R0),@#TBL1   ;TBL1=000377
006740  147037  000000  006536          BICB    @0(R0),@#TBL1   ;TBL1=000252
006746  157037  177776  006537          BISB    @-2(R0),@#TBL1+1                ;TBL1=125252
006754  027037  177774  006536          CMP     @-4(R0),@#TBL1          ;CHECK RESULT
006762  001404                          BEQ     TST115
```

```
3250                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3251                                                    ;           CONDITIONAL BRANCH INST. AND   <====
3252                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
3253                                                    ;           WHICH FOLLOWS W/ 756           <====
3254  006764  012742  000176                  MOV     #176,-(R2)      ;MOVE TO MAILBOX #  *******  176  *******
3255  006770  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3256  006772  000000                            HALT                    ;RESULT OF MODE 7 INSTS INCORRECT
3257                                                                    ;  OR SEQUENCE ERROR
3258
3259                    ;*************************************************************************
3260                    ;
3261                    ;       THIS TEST VERIFIES THE ROTATE MODE 0 INSTRUCTIONS.
3262                    ;R0 IS LOADED WITH A DATA PATTERN, THE C-BIT IS LOADED, AND
3263                    ;AN ROL INSTRUCTION IS EXECUTED WITH MODE 0.  THE OPERATION IS CHECKED
3264                    ;BY TESTING THE RESULTING DATA AND THE STATE OF THE C AND V BITS.
3265                    ;NEXT, THE SAME PROCEDURE IS EXECUTED TO TEST MODE 0 BYTE INSTRUCTIONS.
3266                    ;
3267                    ;*************************************************************************
3268                    ;TEST 115       TEST ROTATE INSTRUCTIONS OF MODE 0
3269                    ;*************************************************************************
3270  006774  005212                TST115: INC     (R2)            ;UPDATE TEST NUMBER
3271  006776  022712  000115                CMP     #115,(R2)       ;SEQUENCE ERROR?
3272  007002  001026                        BNE     TST116-10       ;BR TO ERROR HALT ON SEQ ERROR
3273  007004  012700  125252                MOV     #125252,R0      ;INITIALIZE DATA
3274  007010  000261                        SEC                     ;SET C-BIT
3275  007012  006100                        ROL     R0              ;TRY ROL W/ MODE 0
3276  007014  102004                        BVC     ROT0A           ;CC=0011
3277  007016  103003                        BCC     ROT0A
3279  007020  022700  052525                CMP     #052525,R0      ;CHECK DATA
3279  007024  001404                        BEQ     ROT0B
3280                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3281                                                    ;           CONDITIONAL BRANCH INST. AND   <====
3282                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
3283                                                    ;           WHICH FOLLOWS W/ 767           <====
3284  007026                        ROT0A:
3285  007026  012742  000177                  MOV     #177,-(R2)      ;MOVE TO MAILBOX #  *******  177  *******
3286  007032  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3287  007034  000000                            HALT                    ;ROL MODE 0 FAILED
3288  007036  012700  125252        ROT0B:  MOV     #125252,R0      ;INITIALIZE DATA
3289  007042  000261                        SEC                     ;SET C-BIT
3290  007044  106100                        ROLB    R0              ;TRY ROL W/ MODE 0 EVEN BYTE
3291  007046  102004                        BVC     ROT0C           ;CC=0011
3292  007050  103003                        BCC     ROT0C
3293  007052  022700  125125                CMP     #125125,R0      ;CHECK DATA
3294  007056  001404                        BEQ     TST116
3295                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3296                                                    ;           CONDITIONAL BRANCH INST. AND   <====
3297                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
3298                                                    ;           WHICH FOLLOWS W/ 752           <====
3299  007060                        ROT0C:
3300  007060  012742  000200                  MOV     #200,-(R2)      ;MOVE TO MAILBOX #  *******  200  *******
3301  007064  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3302  007066  000000                            HALT                    ;ROLB MODE 0 FAILED
3303                                                                    ;  OR SEQUENCE ERROR
```

```
                                        ;*****************************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES THE ROTATE MODE 1 INSTRUCTIONS.
                                        ;THE DATA TO BE ROTATED IS IN LOC 0.  R0 IS USED AS THE
                                        ;ADDRESSING REGISTER.  THE C-BIT IS LOADED AND AN ROL IS EXECUTED.
                                        ;THE RESULTS ARE CHECKED BY COMPARING THE DATA RESULTS AND TESTING
                                        ;THE C AND V BITS.  THIS PROCEDURE IS THEN REPEATED TWICE MORE
                                        ;TO TEST THE BYTE ROTATES.  FIRST ON BYTE 0, THEN ON BYTE 1.
                                        ;
                                        ;*****************************************************************************
                                        ;TEST 116       TEST ROTATE INSTRUCTIONS W/ MODE 1
                                        ;*****************************************************************************
007070  005212                TST116: INC     (R2)            ;UPDATE TEST NUMBER
007072  022712  000116              CMP     #116,(R2)       ;SEQUENCE ERROR?
007076  001051                      BNE     TST117-10       ;BR TO ERROR HALT ON SEQ ERROR
007100  005000                      CLR     R0              ;POINT TO LOC. 0
007102  012710  052525              MOV     #52525,(R0)     ;INITIALIZE DATA
007106  000241                      CLC                     ;CLEAR C-BIT
007110  006110                      ROL     (R0)            ;TRY ROL W/ MODE 1
007112  102005                      BVC     ROT1A           ;CC=1010
007114  103404                      BCS     ROT1A
007116  023727  000000  125252      CMP     @#0,#125252     ;CHECK RESULT
007124  001404                      BEQ     ROT1B
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 765           <====
007126                  ROT1A:
007126  012742  000201              MOV     #201,-(R2)      ;MOVE TO MAILBOX #  *******  201  *******
007132  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
007134  000000                      HALT                    ;ROL MODE 1 FAILED
007136  000261                  ROT1B: SEC
007140  012710  125252              MOV     #125252,(R0)    ;INITIALIZE DATA
007144  106110                      ROLB    (R0)            ;TRY ROLB W/ MODE 1 EVEN BYTE
007146  102005                      BVC     ROT1C           ;CC=1011
007150  103004                      BCC     ROT1C
007152  022737  125125  000000      CMP     #125125,@#0     ;TEST RESULT
007160  001404                      BEQ     ROT1D
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 747           <====
007162                  ROT1C:
007162  012742  000202              MOV     #202,-(R2)      ;MOVE TO MAILBOX #  *******  202  *******
007166  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
007170  000000                      HALT                    ;ROLB W/ MODE 1 EVEN BYTE FAILED
007172  012710  125252      ROT1D:  MOV     #125252,(R0)
007176  005000                      CLR     R0              ;POINT TO ODD BYTE
007200  005200                      INC     R0
007202  000261                      SEC                     ;SET C-BIT
007204  106110                      ROLB    (R0)            ;TRY ROLB W/ MODE 1 ODD BYTE
007206  102005                      BVC     ROT1E           ;CC=0011
007210  103004                      BCC     ROT1E
007212  022737  052652  000000      CMP     #052652,@#0     ;CHECK DATA
007220  001404                      BEQ     TST117
```

```
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 727           <====
007222                          ROT1E:
007222  012742  000203                  MOV     #203,-(R2)      ;MOVE TO MAILBOX #  *******  203  *******
007226  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
007230  000000                          HALT                    ;ROLB W/ MODE 1 ODD BYTE FAILED
                                                                ;  OR SEQUENCE ERROR

                                ;*******************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 2 ROTATE INSTRUCTIONS.
                                ;THE SAME PROCEDURE AS IN THE OTHER ROTATE TESTS ARE USED.  R0
                                ;IS USED AS THE ADDRESSING REGISTER AND IS CHECKED FOR PROPER
                                ;INCREMENTING.  BYTE INSTRUCTIONS ARE ALSO CHECKED.
                                ;
                                ;*******************************************************************************
                                ;TEST 117       TEST ROTATE INSTRUCTIONS W/ MODE 2
                                ;*******************************************************************************
007232  005212                  TST117: INC     (R2)            ;UPDATE TEST NUMBER
007234  022712  000117                  CMP     #117,(R2)       ;SEQUENCE ERROR?
007240  001057                          BNE     TST120-10       ;BR TO ERROR HALT ON SEQ ERROR
007242  005000                          CLR     R0              ;POINT TO LOC 0
007244  012710  173737                  MOV     #173737,(R0)    ;INITIALIZE DATA
007250  000241                          CLC                     ;CLEAR C-BIT
007252  006120                          ROL     (R0)+           ;TRY ROL W/ MODE 2
007254  103007                          BCC     ROT2A           ;CHECK C-BIT
007256  022737  167676  000000          CMP     #167676,@#0     ;CHECK DATA
007264  001003                          BNE     ROT2A           ;BRANCH IF RESULT INCORRECT
007266  005300                          DEC     R0              ;TEST R0
007270  005300                          DEC     R0
007272  001404                          BEQ     ROT2B
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 763           <====
007274                          ROT2A:
007274  012742  000204                  MOV     #204,-(R2)      ;MOVE TO MAILBOX #  *******  204  *******
007300  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
007302  000000                          HALT                    ;ROL W/ MODE 2 FAILED
007304  005000                  ROT2B:  CLR     R0              ;POINT TO LOC 0
007306  012710  004040                  MOV     #4040,(R0)      ;INITIALIZE DATA
007312  000241                          CLC                     ;CLEAR C-BIT
007314  106120                          ROLB    (R0)+           ;TRY ROLB W/ MODE 2 EVEN BYTE
007316  103406                          BCS     ROT2C           ;CHECK C-BIT
007320  022737  004100  000000          CMP     #4100,@#0       ;CHECK DATA
007326  001002                          BNE     ROT2C           ;BRANCH IF DATA INCORRECT
007330  005300                          DEC     R0              ;CHECK R0
007332  001404                          BEQ     ROT2D
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 743           <====
007334                          ROT2C:
007334  012742  000205                  MOV     #205,-(R2)      ;MOVE TO MAILBOX #  *******  205  *******
```

```
3416  007340  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3417  007342  000000                          HALT                    ;ROLB W/ MODE 2 EVEN BYTE FAILED
3418  007344  005000                  ROT2D:  CLR     R0              ;POINT TO LOC 0
3419  007346  012710  004040                  MOV     #4040,(R0)      ;INITIALIZE DATA
3420  007352  005200                          INC     R0              ;POINT TO ODD BYTE OF DATA
3421  007354  000261                          SEC                     ;SET C-BIT
3422  007356  106120                          ROLB    (R0)+           ;TRY ROL W/ MODE 2 ODD BYTE
3423  007360  103407                          BCS     ROT2E           ;CHECK C-BIT
3424  007362  022737  010440  000000          CMP     #10440,@#0      ;CHECK DATA
3425  007370  001003                          BNE     ROT2E           ;BRANCH IF DATA INCORRECT
3426  007372  005300                          DEC     R0              ;CHECK R0
3427  007374  005300                          DEC     R0
3428  007376  001404                          BEQ     TST120
3429                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
3430                                                                  ;           CONDITIONAL BRANCH INST. AND    <====
3431                                                                  ;           REPLACE THE MOVE INSTRUCTION    <====
3432                                                                  ;           WHICH FOLLOWS W/ 721            <====
3433  007400                          ROT2E:
3434  007400  012742  000206                  MOV     #206,-(R2)      ;MOVE TO MAILBOX #  *******  206  *******
3435  007404  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3436  007406  000000                          HALT                    ;ROLB W/ MODE 2 ODD BYTE FAILED
3437                                                                  ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 3 ROTATE INSTRUCTIONS.
                                ;THIS TEST USES THE SAME PROCEDURES AS IN THE OTHER ROTATE
                                ;TESTS.  THE DATA IS STORED IN LOC. 0 AND IS ADDRESSED USING
                                ;MODE 37.  BYTE ADDRESSING IS ALSO CHECKED FOR EVEN AND ODD BYTES.
                                ;
                                ;*****************************************************************************
                                ;TEST 120       TEST ROTATE INSTRUCTIONS /W MODE 3
                                ;*****************************************************************************
007410  005212                TST120: INC     (R2)            ;UPDATE TEST NUMBER
007412  022712  000120              CMP     #120,(R2)       ;SEQUENCE ERROR?
007416  001051                      BNE     TST121-10       ;BR TO ERROR HALT ON SEQ ERROR
007420  012737  052525  000000      MOV     #52525,@#0      ;INITIALIZE DATA IN LOC 0
007426  000261                      SEC                     ;SET C-BIT
007430  006137  000000              ROL     @#0             ;TRO ROL W/ MODE 3
007434  103404                      BCS     ROT3A           ;CHECK C-BIT
007436  022737  125253  000000      CMP     #125253,@#0     ;CHECK DATA
007444  001404                      BEQ     ROT3B
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 765           <====
007446                          ROT3A:
007446  012742  000207              MOV     #207,-(R2)      ;MOVE TO MAILBOX #  *******  207  *******
007452  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
007454  000000                      HALT                    ;ROL W/ MODE 3 FAILED
007456  012737  125252  000000  ROT3B: MOV     #125252,@#0     ;INITIALIZE DATA
007464  000241                      CLC                     ;CLEAR C-BIT
007466  106137  000000              ROLB    @#0             ;TRY ROL W/ MODE 3 EVEN BYTE
007472  103004                      BCC     ROT3C           ;CHECK C-BIT
007474  023727  000000  125124  4$:     CMP     @#0,#125124     ;CHECK DATA
007502  001404                      BEQ     ROT3D
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 746           <====
007504                          ROT3C:
007504  012742  000210              MOV     #210,-(R2)      ;MOVE TO MAILBOX #  *******  210  *******
007510  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
007512  000000                      HALT                    ;ROL W/ MODE 3 EVEN BYTE FAILED
007514  012737  125252  000000  ROT3D: MOV     #12525_,@#0     ;INITIALIZE DATA IN LOC. 0
007522  000261                      SEC                     ;SET C-BIT
007524  106137  000001              ROLB    @#1             ;TRY ROL W/ MODE 3 ODD BYTE
007530  103004                      BCC     ROT3E           ;CHECK C-BIT
007532  022737  052652  000000      CMP     #052652,@#0     ;CHECK DATA
007540  001404                      BEQ     TST121
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 727           <====
007542                          ROT3E:
007542  012742  000211              MOV     #211,-(R2)      ;MOVE TO MAILBOX #  *******  211  *******
007546  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
007550  000000                      HALT                    ;ROL W/ MODE 3 ODD BYTE FAILED
```

```
3494                                                    ;  OR SEQUENCE ERROR
```

```
3495
3496                                    ;*****************************************************************************
3497                                    ;
3498                                    ;       THIS TEST VERIFIES MODE 4 ROTATE INSTRUCTIONS.  THE DATA IS
3499                                    ;STORED IN LOC. 0.  R0 IS SET TO 2 AND THE CARRY IS SET.  AN ROL MODE 4
3500                                    ;IS USED TO ROTATE LOCATION 0 USING R0.  THE DATA IS CHECKED
3501                                    ;AND THE C AND V BITS ARE TESTED.  THE PROPER DECREMENTING OF
3502                                    ;R0 IS VERIFIED.
3503                                    ;
3504                                    ;*****************************************************************************
3505                                    ;TEST 121       TEST MODE 4 W/ ROTATE INSTRUCTIONS
3506                                    ;*****************************************************************************
3507  007552  005212                   TST121: INC     (R2)            ;UPDATE TEST NUMBER
3508  007554  022712  000121                   CMP     #121,(R2)       ;SEQUENCE ERROR?
3509  007560  001016                           BNE     TST122-10       ;BR TO ERROR HALT ON SEQ ERROR
3510  007562  012737  070707  000000           MOV     #070707,@#0     ;INITIALIZE DATA IN LOC. 0
3511  007570  012700  000002                   MOV     #2,R0           ;INITIALIZE R0 AS POINTER
3512  007574  000261                           SEC                     ;SET C-BIT
3513  007576  006140                           ROL     -(R0)           ;TRY ROL W/ MODE 4
3514  007600  103406                           BCS     ROT4            ;CHECK C-BIT
3515  007602  022737  161617  000000           CMP     #161617,@#0     ;CHECK DATA
3516  007610  001002                           BNE     ROT4            ;BRANCH IF DATA INCORRECT
3517  007612  005700                           TST     R0              ;CHECK MODE 4 REGISTER
3518  007614  001404                           BEQ     TST122
3519                                                                   ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
3520                                                                   ;           CONDITIONAL BRANCH INST. AND    <====
3521                                                                   ;           REPLACE THE MOVE INSTRUCTION    <====
3522                                                                   ;           WHICH FOLLOWS W/ 762            <====
3523  007616                           ROT4:
3524  007616  012742  000212                   MOV     #212,-(R2)      ;MOVE TO MAILBOX # \ *******  212  *******
3525  007622  005242                           INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3526  007624  000000                           HALT                    ;ROL MODE 4 FAILED
3527                                                                   ;  OR SEQUENCE ERROR
3528
3529                                    ;*****************************************************************************
3530                                    ;
3531                                    ;       THIS TEST VERIFIES MODE 5 ROTATE INSTRUCTIONS.
3532                                    ;THE DATA IS STORED IN A WORK LOCATION (ROTX) AT THE END OF THE
3533                                    ;TEST CODE.  LOC. 0 IS LOADED WITH THE ADDRESS OF THE DATA (ROTX).
3534                                    ;R0 IS SET TO 2.  THE CARRY IS CLEARED AND A MODE 5 ROL
3535                                    ;IS EXECUTED USING R0 AS AN ADDRESSING REGISTER.  THE DATA IS
3536                                    ;CHECKED, THE C AND V BITS TESTED, AND R0 CHECKED FOR PROPER
3537                                    ;DECREMENTING.
3538                                    ;
3539                                    ;*****************************************************************************
3540                                    ;TEST 122       TEST MODE 5 W/ ROTATE INSTRUCTIONS
3541                                    ;*****************************************************************************
3542  007626  005212                   TST122: INC     (R2)            ;UPDATE TEST NUMBER
3543  007630  022712  000122                   CMP     #122,(R2)       ;SEQUENCE ERROR?
3544  007634  001021                           BNE     ROT5            ;BR TO ERROR HALT ON SEQ ERROR
3545  007636  012737  007710  000000           MOV     #ROTX,@#0       ;MOVE POINTER TO LOC. 0
3546  007644  012700  000002                   MOV     #2,R0           ;SET MODE 5 REG. TO LOC. 0
3547  007650  012767  107070  000032           MOV     #107070,ROTX    ;INITIALIZE DATA
3548  007656  000241                           CLC                     ;CLEAR C-BIT
3549  007660  006150                           ROL     @-(R0)          ;TRY ROL W/ MODE 5
3550  007662  103006                           BCC     ROT5            ;CHECK C-BIT
```

```
3551  007664  022737  016160  007710          CMP     #016160,@#ROTX  ;CHECK DATA
3552  007672  001002                          BNE     ROT5            ;BRANCH IF DATA INCORRECT
3553  007674  005700                          TST     R0              ;CHECK MODE 5 REGISTER
3554  007676  001405                          BEQ     TST123
3555                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3556                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
3557                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
3558                                                                  ;           WHICH FOLLOWS W/ 757           <====
3559  007700                          ROT5:
3560  007700  012742  000213                  MOV     #213,-(R2)      ;MOVE TO MAILBOX #  *******  213  *******
3561  007704  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3562  007706  000000                          HALT                    ;ROL MODE 5 FAILED
3563                                                                  ;  OR SEQUENCE ERROR
3564  007710  000000                  ROTX:   0
3565
3566                                  ;*******************************************************************************
3567                                  ;
3568                                  ;       THIS TEST VERIFIES MODE 6 ROTATE INSTRUCTIONS.
3569                                  ;IT USES THE SAME PROCEDURE AS THE ABOVE TEST EXCEPT THE
3570                                  ;ROTATE INSTRUCTION USES MODE 6 ADDRESSING WITH REGISTER 7.
3571                                  ;THE DATA IS STILL OPERATED ON IN LOC. ROTX (SEE PREVIOUS TEST).
3572                                  ;
3573                                  ;*******************************************************************************
3574                                  ;TEST 123       TEST MODE 6 W/ ROTATE INSTRUCTIONS
3575                                  ;*******************************************************************************
3576  007712  005212                  TST123: INC     (R2)            ;UPDATE TEST NUMBER
3577  007714  022712  000123                  CMP     #123,(R2)       ;SEQUENCE ERROR?
      007720  001013                          BNE     TST124-10       ;BR TO ERROR HALT ON SEQ ERROR
      007722  012737  125252  007710          MOV     #125252,@#ROTX  ;INITIALIZE DATA
3580  007730  000261                          SEC                     ;SET C-BIT
3581  007732  006167  177752                  ROL     ROTX            ;TRY ROL W/ MODE 6
3582  007736  103004                          BCC     ROT6            ;CHECK C-BIT
3583  007740  022737  052525  007710          CMP     #52525,@#ROTX   ;CHECK DATA
3584  007746  001404                          BEQ     TST124
3585                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3586                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
3587                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
3588                                                                  ;           WHICH FOLLOWS W/ 765           <====
3589  007750                          ROT6:
3590  007750  012742  000214                  MOV     #214,-(R2)      ;MOVE TO MAILBOX #  *******  214  *******
3591  007754  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3592  007756  000000                          HALT                    ;ROL W/ MODE 6 FAILED
3593                                                                  ;  OR SEQUENCE ERROR
```

```
                                        ;*****************************************************************************
                                        ;
                                        ;THIS TEST VERIFIES MODE 7 ROTATE INSTRUCTIONS.
                                        ;THE DATA IS SET IN LOC. ROTX, (SEE PREVIOUS TEST).  THE ROL INSTRUCTION
                                        ;ADDRESSES IT INDIRECTLY USING MODE 7 AND INDIRECT ADDRESS LOCATION
                                        ;(ROTXAD) FOLLOWING THE TEST CODE.
                                        ;
                                        ;*****************************************************************************
                                        ;TEST 124       TEST MODE 7 W/ ROTATE INSTRUCTIONS
                                        ;*****************************************************************************
010000  007760  005212                  TST124: INC     (R2)            ;UPDATE TEST NUMBER
010002  007762  022712  000124                  CMP     #124,(R2)       ;SEQUENCE ERROR?
010006  007766  001016                          BNE     ROT7            ;BR TO ERROR HALT ON SEQ ERROR
010010  007770  012737  052525  007710          MOV     #52525,@#ROTX   ;INITIALIZE DATA
010016  007776  012737  007710  010034          MOV     #ROTX,@#ROTXAD  ;INITIALIZE ADDRESS POINTER
010024  010004  000241                          CLC                     ;CLEAR C-BIT
010026  010006  006177  000022                  ROL     @ROTXAD         ;TRY ROL W/ MODE 7
010032  010012  103404                          BCS     ROT7            ;CHECK C-BIT
010034  010014  023727  007710  125252          CMP     @#ROTX,#125252  ;CHECK DATA
010042  010022  001405                          BEQ     TST125
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                        ;           WHICH FOLLOWS W/ 762           <====
010044                                  ROT7:
010044  012742  000215                          MOV     #215,-(R2)      ;MOVE TO MAILBOX #  *******  215  *******
010050  005242                                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
010052  000000                                  HALT                    ;ROL W/ MODE 7 FAILED
                                                                        ;  OR SEQUENCE ERROR
010054  000000                          ROTXAD: 0


                                        ;*****************************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES MODE 0 SWAB INSTRUCTION.  R0 IS SET TO
                                        ;177400.  A SWAB MODE 0 IS EXECUTED AND THE CONDITIONAL BRANCH
                                        ;IS USED TO CHECK THE SIGN OF THE RESULT.  ALSO, A COMPARISON
                                        ;IS MADE TO CHECK THE DATA RESULTS.
                                        ;
                                        ;*****************************************************************************
                                        ;TEST 125       TEST MODE 0 W/ SWAB INST.
                                        ;*****************************************************************************
010056  010036  005212                  TST125: INC     (R2)            ;UPDATE TEST NUMBER
010060  010040  022712  000125                  CMP     #125,(R2)       ;SEQUENCE ERROR?
010064  010044  001013                          BNE     TST126-10       ;BR TO ERROR HALT ON SEQ ERROR
010066  010046  012700  177400                  MOV     #177400,R0      ;MOVE TEST PATTERN TO R0
010072  010052  000300                          SWAB    R0              ;TRY SWAB MODE 0
010074  010054  100404                          BMI     SB0
                                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                        ;           WHICH FOLLOWS W/ 774           <====
010076  010056  012742  000216                  MOV     #216,-(R2)      ;MOVE TO MAILBOX #  *******  216  *******
010102  010062  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
010104  010064  000000                          HALT                    ;SWAB DID NOT SET CC'S CORRECT
```

```
3650  010066  022700  000377          SB5:    CMP     #377,R0         ;CHECK RESULT
3651  010072  001404                          BEQ     TST126
3652                                          ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3653                                          ;           CONDITIONAL BRANCH INST. AND   <====
3654                                          ;           REPLACE THE MOVE INSTRUCTION   <====
3655                                          ;           WHICH FOLLOWS w/ 765           <====
3656  010074  012742  000217                  MOV     #217,-(R2)      ;MOVE TO MAILBOX #  *******  217  *******
3657  010100  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3658  010102  000000                          HALT                    ;RESULT OF SWAB MODE 5 FAILED
3659                                                                  ;  OR SEQUENCE ERROR
```

```
                                        ;*****************************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES MODE 1 SWAB INSTRUCTION.  THE TEST
                                        ;PATTERN IS MOVED TO LOC 0.  R0 IS CLEARED AND USED AS THE ADDRESSING
                                        ;REGISTER IN THE MODE 1 SWAB.  THE DATA RESULTS ARE CHECKED WITH
                                        ;A COMPARE.
                                        ;
                                        ;*****************************************************************************
                                        ;TEST 126       TEST MODE 1 W/ SWAB INST
                                        ;*****************************************************************************
010104  005212                  TST126: INC     (R2)            ;UPDATE TEST NUMBER
010106  022712  000126                  CMP     #126,(R2)       ;SEQUENCE ERROR?
010112  001011                          BNE     TST127-10       ;BR TO ERROR HALT ON SEQ ERROR
010114  012737  125652  000000          MOV     #125652,@#0     ;MOVE TEST PATTERN TO LOC. 0
010122  005000                          CLR     R0              ;R0=0
010124  000310                          SWAB    (R0)            ;TRY SWAB MODE 1
010126  022737  125253  000000          CMP     #125253,@#0     ;CHECK RESULT
010134  001404                          BEQ     TST127
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 767           <====
010136  012742  000220                  MOV     #220,-(R2)      ;MOVE TO MAILBOX #  *******  220  *******
010142  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
010144  000000                          HALT                    ;RESULT OF SWAB MODE 1 FAILED
                                                                ;  OR SEQUENCE ERROR


                                        ;*****************************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES MODE 2 SWAB INSTRUCTION.  THE TEST
                                        ;PATTERN IS MOVED TO LOC 0.  R0 IS CLEARED AND USED AS THE MODE
                                        ;2 ADDRESSING REGISTER.  THE RESULTS ARE CHECKED WITH A COMPARE.
                                        ;R0 IS CHECKED FOR PROPER DECREMENTING.
                                        ;
                                        ;*****************************************************************************
                                        ;TEST 127       TEST MODE 2 W/ SWAB INST
                                        ;*****************************************************************************
010146  005212                  TST127: INC     (R2)            ;UPDATE TEST NUMBER
010150  022712  000127                  CMP     #127,(R2)       ;SEQUENCE ERROR?
010154  001020                          BNE     TST130-10       ;BR TO ERROR HALT ON SEQ ERROR
010156  012737  125152  000000          MOV     #125152,@#0     ;MOVE TEST PATTERN TO LOC. 0
010164  005000                          CLR     R0              ;R0=0
010166  000320                          SWAB    (R0)+           ;TRY SWAB MODE 2
010170  022737  065252  000000          CMP     #65252,@#0      ;CHECK RESULT
010176  001404                          BEQ     SB2
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 767           <====
010200  012742  000221                  MOV     #221,-(R2)      ;MOVE TO MAILBOX #  *******  221  *******
010204  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
010206  000000                          HALT                    ;RESULT OF SWAB MODE 0 FAILED
010210  162700  000002          SB2:    SUB     #2,R0           ;CHECK EFFECT OF REG.
010214  001404                          BEQ     TST130
```

```
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                        ;           WHICH FOLLOWS W/ 760            <====
3720  010216  012742  000222          MOV     #222,-(R2)  ;MOVE TO MAILBOX #  *******  222  *******
3721  010222  005242                  INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
3722  010224  000000                  HALT                ;REGISTER VALUE INCORRECT
                                                        ;  OR SEQUENCE ERROR
3723
3724
3725
3726                                ;*************************************************************************
3727                                ;
3728                                ;       THIS TEST VERIFIES MODE 3 SWAB INSTRUCTION.  THE TEST
3729                                ;PATTERN IS MOVED TO LOC 0.  A MODE 3 SWAB INSTRUCTION IS EXECUTED
3730                                ;USING R7 AS THE ADDRESSING REGISTER.  A COMPARE VERIFIES THE
3731                                ;DATA RESULTS.
3732                                ;
3733                                ;*************************************************************************
3734                                ;TEST 130       TEST MODE 3 W/SWAB INST.
3735                                ;*************************************************************************
3736  010226  005212                TST130: INC     (R2)        ;UPDATE TEST NUMBER
3737  010230  022712  000130                CMP     #130,(R2)   ;SEQUENCE ERROR?
3738  010234  001011                        BNE     TST131-10   ;BR TO ERROR HALT ON SEQ ERROR
3739  010236  012737  000377  000000        MOV     #377,@#0    ;MOVE TEST PATTERN TO LOC. 0
3740  010244  000337  000000                SWAB    @#0         ;TRY SWAB W/ MODE 3
3741  010250  022737  177400  000000        CMP     #177400,@#0 ;CHECK RESULT
3742  010256  001404                        BEQ     TST131
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                        ;           CONDITIONAL BRANCH INST. AND    <====
                                                        ;           REPLACE THE MOVE INSTRUCTION    <====
                                                        ;           WHICH FOLLOWS W/ 767            <====
3747  010260  012742  000223          MOV     #223,-(R2)  ;MOVE TO MAILBOX #  *******  223  *******
3748  010264  005242                  INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
3749  010266  000000                  HALT                ;RESULT OF SWAB INCORRECT
3750                                                    ;  OR SEQUENCE ERROR
```

```
3751
3752
3753                                    ;*****************************************************************************
3754                                    ;
3755                                    ;       THIS TEST VERIFIES MODE 4 SWAB INSTRUCTIONS.  THE DATA
3756                                    ;IS MOVED TO LOC 0.  R0 IS SET TO 2 AND USED AS THE MODE 4 ADDRESSING
3757                                    ;REGISTER.  THE DATA IS CHECKED WITH A COMPARE AND R0 IS CHECKED
3758                                    ;FOR PROPER DECREMENTING.
3759                                    ;
3760                                    ;*****************************************************************************
3761                                    ;TEST 131       TEST MODE 4 W/ SWAB INST
3762                                    ;*****************************************************************************
3763  010270  005212                    TST131: INC     (R2)                    ;UPDATE TEST NUMBER
3764  010272  022712  000131                    CMP     #131,(R2)               ;SEQUENCE ERROR?
3765  010276  001020                            BNE     TST132-10               ;BR TO ERROR HALT ON SEQ ERROR
3766  010300  012737  125652  000000            MOV     #125652,@#0             ;MOVE TEST PATTERN TO LOC. 0
3767  010306  012700  000002                    MOV     #2,R0                   ;SET UP REGISTER POINTER
3768  010312  000340                            SWAB    -(R0)                   ;TRY SWAB MODE 4
3769  010314  022737  125253  000000            CMP     #125253,@#0             ;CHECK RESULT
3770  010322  001404                            BEQ     SB4
3771                                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3772                                                                            ;           CONDITIONAL BRANCH INST. AND   <====
3773                                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
3774                                                                            ;           WHICH FOLLOWS W/ 766           <====
3775  010324  012742  000224                    MOV     #224,-(R2)              ;MOVE TO MAILBOX #  *******  224  *******
3776  010330  005242                            INC     -(R2)                   ;SET MSGTYP TO FATAL ERROR
3777  010332  000000                            HALT                            ;RESULT OF SWAB INCORRECT
3778  010334  005700                    SB4:    TST     R0                      ;CHECK EFFECT ON REG.
3779  010336  001404                            BEQ     TST132
3780                                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3781                                                                            ;           CONDITIONAL BRANCH INST. AND   <====
3782                                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
3783                                                                            ;           WHICH FOLLOWS W/ 760           <====
3784  010340  012742  000225                    MOV     #225,-(R2)              ;MOVE TO MAILBOX #  *******  225  *******
3785  010344  005242                            INC     -(R2)                   ;SET MSGTYP TO FATAL ERROR
3786  010346  000000                            HALT                            ;REGISTER VALUE INCORRECT
3787                                                                            ;  OR SEQUENCE ERROR
3789
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 5 SWAB INSTRUCTION.  THE TEST USES
                                ;TWO LOCATIONS FOLLOWING THE TEST CODE.  SB5X HOLDS THE DATA;
                                ;SB5XAD IS A POINTER TO THE DATA LOCATION.  THE DATA IS MOVED TO
                                ;SB5X AND R0 IS SET TO TWO PLUS THE ADDRESS OF SB5XAD   FOLLOWING
                                ;THE MODE 5 SWAB SB5X IS CHECKED FOR THE PROPER DATA.  R0 IS
                                ;CHECKED TO SEE THAT IT WAS DECREMENTED PROPERLY.
                                ;
                                ;*****************************************************************************
                                ;TEST 132       TEST MODE 5 W/ SWAB INST.
                                ;*****************************************************************************
010350  005212          TST132: INC     (R2)            ;UPDATE TEST NUMBER
010352  022712  000132          CMP     #132,(R2)       ;SEQUENCE ERROR?
010356  001021                  BNE     SB5             ;BR TO ERROR HALT ON SEQ ERROR
010360  012700  010436          MOV     #SB5XAD+2,R0    ;SET UP POINTER TO WORK LOCATION
010364  012767  125125  000040  MOV     #125125,SB5X    ;MOVE PATTERN TO WORK LOCATION
010372  000350                  SWAB    @-(R0)          ;TRY SWAB MODE 5
010374  022767  052652  000030  CMP     #52652,SB5X     ;CHECK RESULT
010402  001404                  BEQ     SB5A
                                ;  TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                ;            CONDITIONAL BRANCH INST. AND   <====
                                ;            REPLACE THE MOVE INSTRUCTION   <====
                                ;            WHICH FOLLOWS W/ 766           <====
010404  012742  000226          MOV     #226,-(R2)      ;MOVE TO MAILBOX #  *******  226  *******
010410  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
010412  000000                  HALT                    ;RESULT OF SWAB INCORRECT
010414  020027  010434  SB5A:   CMP     R0,#SB5XAD      ;CHECK RESULT OF REG.
010420  001406                  BEQ     TST133
                                ;  TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                ;            CONDITIONAL BRANCH INST. AND   <====
                                ;            REPLACE THE MOVE INSTRUCTION   <====
                                ;            WHICH FOLLOWS W/ 757           <===-
010422          SB5:
010422  012742  000227          MOV     #227,-(R2)      ;MOVE TO MAILBOX #  *******  227  *******
010426  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
010430  000000                  HALT                    ;REGISTER VALUE INCORRECT
                                ;  OR SEQUENCE ERROR
010432  000000          SB5X:   0                       ;WORK LOCATION
010434  010432          SB5XAD: SB5X
```

```
3831
3832                            ;*****************************************************************************
3833                            ;
3834                            ;       THIS TEST VERIFIES MODE 6 SWAB INSTRUCTION.  THIS TEST
3835                            ;USES A WORK LOCATION (SB6X) FOLLOWING THE TEST CODE.  TEST DATA
3836                            ;IS LOADED INTO THE WORK LOCATION.  R0, THE ADDRESSING REGISTER
3837                            ;IS LOADED WITH 6 LESS THEN THE ADDRESS OF THE WORK LOCATION.
3838                            ;THE MODE 6 SWAB IS EXECUTED WITH A +6 OFFSET.  THE DATA IS
3839                            ;VERIFIED WITH A COMPARE.
3840                            ;
3841                            ;*****************************************************************************
3842                            ;TEST 133       TEST MODE 6 W/ SWAB INST.
3843                            ;*****************************************************************************
3844  010436  005212            TST133: INC     (R2)            ;UPDATE TEST NUMBER
3845  010440  022712  000133            CMP     #133,(R2)       ;SEQUENCE ERROR?
3846  010444  001013                    BNE     SB6             ;BR TO ERROR HALT ON SEQ ERROR
3847  010446  012767  125125  000030    MOV     #125125,SB6X    ;MOVE PATTERN TO WORK LOCATION
3848  010454  012700  010476            MOV     #SB6X-6,R0      ;MOVE OFFSET POINTER TO R0
3849  010460  000360  000006            SWAB    6(R0)           ;TRY SWAB W/ MODE 6
3850  010464  022760  052652  000006    CMP     #52652,6(R0)    ;CHECK RESULT
3851  010472  001405                    BEQ     TST134
3852                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3853                                                            ;           CONDITIONAL BRANCH INST. AND   <====
3854                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
3855                                                            ;           WHICH FOLLOWS W/ 765           <====
3856  010474                    SB6:
3857  010474  012742  000230            MOV     #230,-(R2)      ;MOVE TO MAILBOX #  *******  230  *******
3858  010500  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3859  010502  000000                    HALT                    ;RESULT OF SWAB INCORRECT
3860                                                            ;  OR SEQUENCE ERROR
3861  010504  000000            SB6X:   0                       ;WORK LOCATION
3862
```

```
                                ;*******************************************************************
                                ;
                                ;       THIS TEST VERIFIES MODE 7 SWAB INSTRUCTION.  THIS TEST
                                ;USES TWO LOCATIONS FOLLOWING THE TEST CODE:  A WORK LOCATION
                                ;(SB7X) AND A POINTER TO THE WORK LOCATION (SB7XAD).  DATA IS MOVED
                                ;TO THE WORK LOCATION.  RO IS LOADED WITH 72 LESS THAN THE ADDRESS
                                ;OF THE ADDRESS POINTER.  THE DATA IS SWAB'ED USING A MODE 7
                                ;INSTRUCTION WITH AN OFFSET OF +72.  THE DATA IS VERIFIED WITH A
                                ;COMPARE.
                                ;
                                ;*******************************************************************
                                ;TEST 134       TEST MODE 7 W/ SWAB INST.
                                ;*******************************************************************
010506  005212                  TST134: INC     (R2)            ;UPDATE TEST NUMBER
010510  022712  000134                  CMP     #134,(R2)       ;SEQUENCE ERROR?
010514  001013                          BNE     SB7             ;BR TO ERROR HALT ON SEQ ERROR
010516  012767  177400  000030          MOV     #177400,SB7X    ;MOVE PATTERN TO WORK LOCATION
010524  012700  010464                  MOV     #SB7XAD-72,RO   ;MOVE OFFSET POINTER TO RO
010530  000370  000072                  SWAB    @72(RO)         ;TRY SWAB MODE 7
010534  027027  000072  000377          CMP     @72(RO),#377    ;CHECK RESULTS
010542  001406                          BEQ     TST135
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS      <====
                                                                ;           CONDITIONAL BRANCH INST. AND      <====
                                                                ;           REPLACE THE MOVE INSTRUCTION      <====
                                                                ;           WHICH FOLLOWS W/ 765              <====
010544                          SB7:
010544  012742  000231                  MOV     #231,-(R2)      ;MOVE TO MAILBOX #  *******  231  *******
010550  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
010552  000000                          HALT                    ;RESULT OF SWAB INCORRECT
                                                                ;  OR SEQUENCE ERROR
010554  000000                  SB7X:   0                       ;WORK LOCATION
010556  010554                  SB7XAD: SB7X                    ;POINTER TO WORK LOCATION
```

```
3897
3898                    ;*****************************************************************************
3899                    ;
3900                    ;       THIS TEST VERIFIES ALL LEGAL MODES OF THE JMP INSTRUCTION.
3901                    ;BECAUSE OF THE NATURE OF THE INSTRUCTION UNDER TEST, THIS TEST
3902                    ;UTILIZES SEVERAL DIFFERENT TECHNIQUES.  THE CODE IS NOT EXECUTED
3903                    ;IN A LINEAR FASHION.  THE DIFFERENT MODES ARE EXECUTED IN ORDER
3904                    ;FROM 1-7; HOWEVER, THE CODE IS ARRANGED SO THAT CONTROL LEAP
3905                    ;FROGS THRU THE TEST CODE.  THE ORDER OF APPEARANCE OF THE CODE
3906                    ;IS:
3907                    ;       JMP MODE 1
3908                    ;       JMP MODE 3
3909                    ;       JMP MODE 2
3910                    ;       JMP MODE 4
3911                    ;       JMP MODE 6
3912                    ;       JMP MODE 5
3913                    ;       JMP MODE 7
3914                    ;AN INTERNAL SEQUENCE TEST (JMPSEQ) IS USED TO INSURE THAT THE
3915                    ;JUMPS ARE OCCURRING IN THE PROGRAMMED SEQUENCE.
3916                    ;       THE TEST IS MADE UP OF SEVERAL BLOCKS OF CODE.  EACH CODE
3917                    ;BEGINS WITH A LABEL WHICH INDICATES THE MODE BEING EXECUTED IN
3918                    ;THAT BLOCK.  A SIMPLE PROCEDURE IS FOLLOWED IN EACH BLOCK.  FOR
3919                    ;EXAMPLE THE CODE BEGINNING AT JMP3 WILL FIRST COMPARE THE RESULTS
3920                    ;OF THE PREVIOUS MODE 2 JUMP.  (ANY REGISTER CHANGES ARE VERIFIED
3921                    ;AND THE SEQUENCE CHECK IS MADE).  THEN THE REGISTERS ARE SETUP
3922                    ;FOR A MODE 3 JUMP TO THE NEXT TEST BLOCK (HERE, JMP4), THE SEQUENCE
3923                    ;CHECKER IS UPDATED AND THE JUMP IS EXECUTED.
3924                    ;       IF A FAILURE OCCURS, THE SEQUENCE CHECKER WILL ASSIST IN
3925                    ;DETERMINING JUST WHICH MODE FAILED.  IF THE SEQUENCE IS CORRECT
3926                    ;THEN THE ERROR DETECTED WAS A MODE FAILURE (E.G. FAILURE OF THE
3927                    ;REGISTER TO BE INCREMENTED IN MODE 2 JUMP.)
3928                    ;
3929                    ;*****************************************************************************
3930                    ;TEST 135       TEST THE JMP INSTRUCTION IN ALL MODES
3931                    ;*****************************************************************************
3932  010560  005212    TST135: INC     (R2)            ;UPDATE TEST NUMBER
3933  010562  022712  000135    CMP     #135,(R2)       ;SEQUENCE ERROR?
3934  010566  001150            BNE     JMPCK+6         ;BR TO ERROR HALT ON SEQ ERROR
3935  010570  005067  000326    CLR     JMPSEQ          ;ESTABLISH A SEQUENCE CHECKER
3936  010574  012700  010654    MOV     #JMP2,R0        ;SET R0=JUMP TARGET
3937  010600  000110            JMP     (R0)            ;TRY JMP MODE 1
3938  010602  022700  010604    JMP3:   CMP     #.+2,R0         ;CHECK RESULT OF MODE 2 JUMP
3939  010606  001404            BEQ     JMP3A
3940                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3941                                                    ;           CONDITIONAL BRANCH INST. AND   <====
3942                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
3943                                                    ;           WHICH FOLLOWS W/ 770           <====
3944  010610  012742  000232            MOV     #232,-(R2)      ;MOVE TO MAILBOX #  *******  232  *******
3945  010614  005242            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
3946  010616  000000            HALT                    .
3947  010620  026727  000276  000001  JMP3A:  CMP     JMPSEQ,#1       ;MAKE SURE JMPS ARE IN SEQUENCE: JMPSEQ=1?
3948  010626  001404            BEQ     JMP3B
3949                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
3950                                                    ;           CONDITIONAL BRANCH INST. AND   <====
3951                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
3952                                                    ;           WHICH FOLLOWS W/ 760           <====
```

```
 3953  010630  012742  000233                 MOV     #233,-(R2)      ;MOVE TO MAILBOX #  *******  233  *******
 3954  010634  005242                         INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 3955  010636  000000                         HALT                    ;SHOULD BE HERE FROM JMP MODE 2 ONLY
 3956  010640  012700  010652       JMP3B:   MOV     #IJMP4,R0       ;POINT R0 TO INDIRECT JMP ADDR.
 3957  010644  005267  000252                 INC     JMPSEQ          ;UPDATE SEQUENCE CHECKER
 3958  010650  000130                         JMP     @(R0)+          ;TRY JMP MODE 3
 3959  010652  010704             IJMP4:      JMP4                    ;ADDRESS INDIRECT JUMP
 3960
 3961  010654  005767  000242       JMP2:    TST     JMPSEQ          ;CHECK THAT JMPS ARE IN SEQUENCE: JMPSEQ=0?
 3962  010660  001404                         BEQ     JMP2A
 3963                                                                 ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 3964                                                                 ;           CONDITIONAL BRANCH INST. AND   <====
 3965                                                                 ;           REPLACE THE MOVE INSTRUCTION   <====
 3966                                                                 ;           WHICH FOLLOWS W/ 743           <====
 3967  010662  012742  000234                 MOV     #234,-(R2)      ;MOVE TO MAILBOX #  *******  234  *******
 3968  010666  005242                         INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 3969  010670  000000                         HALT                    ;SHOULD BE HERE FROM JMP MODE 1 ONLY
 3970  010672  005267  000224       JMP2A:   INC     JMPSEQ          ;UPDATE SEQUENCE CHECKER
 3971  010676  012700  010602                 MOV     #JMP3,R0        ;SET R0=JUMP TARGET
 3972  010702  000120                         JMP     (R0)+           ;TRY A JUMP MODE 2 TO "JMP3"
 3973  010704  022700  010654       JMP4:    CMP     #IJMP4+2,R0     ;CHECK RESULT OF REGISTER IN MODE 3 JUMP
 3974  010710  001404                         BEQ     JMP4A
 3975                                                                 ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 3976                                                                 ;           CONDITIONAL BRANCH INST. AND   <====
 3977                                                                 ;           REPLACE THE MOVE INSTRUCTION   <====
 3978                                                                 ;           WHICH FOLLOWS W/ 727           <====
 3979  010712  012742  000235                 MOV     #235,-(R2)      ;MOVE TO MAILBOX #  *******  235  *******
 3980  010716  005242                         INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 3981  010720  000000                         HALT                    ;REGISTER VALUE AFTER MODE 3 JUMP INCORRECT
 3982  010722  022767  000002  000172 JMP4A:  CMP     #2,JMPSEQ       ;CHECK JUMP SEQUENCE: JMPSEQ=2?
 3983  010730  001404                         BEQ     JMP4B
 3984                                                                 ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 3985                                                                 ;           CONDITIONAL BRANCH INST. AND   <====
 3986                                                                 ;           REPLACE THE MOVE INSTRUCTION   <====
 3987                                                                 ;           WHICH FOLLOWS W/ 717           <====
 3988  010732  012742  000236                 MOV     #236,-(R2)      ;MOVE TO MAILBOX #  *******  236  *******
 3989  010736  005242                         INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 3990  010740  000000                         HALT                    ;SHOULD BE ONLY FROM MODE 3 JUMP
 3991  010742  012700  011012       JMP4B:   MOV     #JMP5+2,R0      ;SET UP POINTER TO JUMP TARGET
 3992  010746  005267  000150                 INC     JMPSEQ          ;UPDATE SEQUENCE CHECKER
 3993  010752  000140                         JMP     -(R0)           ;TRY JUMP MODE 4 TO "JMP4"
 3994
 3995  010754  022767  000004  000140 JMP6:   CMP     #4,JMPSEQ       ;CHECK THAT JUMPS ARE IN SEQUENCE: JMPSEQ=4?
 3996  010762  001404                         BEQ     JMP6A
 3997                                                                 ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 3998                                                                 ;           CONDITIONAL BRANCH INST. AND   <====
 3999                                                                 ;           REPLACE THE MOVE INSTRUCTION   <====
 4000                                                                 ;           WHICH FOLLOWS W/ 702           <====
 4001  010764  012742  000237                 MOV     #237,-(R2)      ;MOVE TO MAILBOX #  *******  237  *******
 4002  010770  005242                         INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 4003  010772  000000                         HALT                    ;SHOULD BE HERE ONLY FROM MODE 5 JUMP
 4004  010774  012700  011442       JMP6A:   MOV     #JMP7+376,R0    ;SET UP OFFSET POINTER TO JUMP TARGET
 4005  011000  005267  000116                 INC     JMPSEQ          ;UPDATE JUMP SEQUENCE
 4006  011004  000160  177402                 JMP     -376(R0)        ;TRY MODE 6 JUMP
 4007
 4008  011010  022767  000003  000104 JMP5:   CMP     #3,JMPSEQ       ;CHECK THAT JUMPS ARE IN SEQUENCE: JMPSEQ=3?
```

```
4009  011016  001404                          BEQ     JMP5A
4010                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4011                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4012                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4013                                                  ;           WHICH FOLLOWS W/ 664           <====
4014  011020  012742  000240                        MOV     #240,-(R2)  ;MOVE TO MAILBOX #  *******  240  *******
4015  011024  005242                                INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
4016  011026  000000                                HALT                ;SHOULD ONLY BE HERE FROM MODE 4 JUMP
4017  011030  012700  011044          JMP5A:  MOV     #IJMP5+2,R0 ;SET UP POINTER TO INDIRECT JUMP ADDR.
4018  011034  005267  000062                        INC     JMPSEQ      ;UPDATE JUMP SEQUENCE
4019  011040  000150                                JMP     @-(R0)      ;TRY JUMP MODE 5 TO "JMP6"
4020  011042  010754                  IJMP5:  JMP6                ;INDIRECT ADDRESS POINTER
4021
4022  011044  022767  000005  000050  JMP7:   CMP     #5,JMPSEQ   ;CHECK JUMPS IN SEQUENCE: JMPSEQ=5?
4023  011052  001404                          BEQ     JMP7A
4024                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4025                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4026                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4027                                                  ;           WHICH FOLLOWS W/ 646           <====
4028  011054  012742  000241                        MOV     #241,-(R2)  ;MOVE TO MAILBOX #  *******  241  *******
4029  011060  005242                                INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
4030  011062  000000                                HALT                ;SHOULD ONLY BE HERE FROM MODE 6 JUMP
4031  011064  012700  011110          JMP7A:  MOV     #IJMP+10,R0 ;SET UP OFFSET POINTER TO INDIRECT ADDR.
4032  011070  005267  000026                        INC     JMPSEQ      ;UPDATE JUMP SEQUENCE
4033  011074  000170  177770                        JMP     @-10(R0)    ;TRY MODE 7 JUMP
4034  011100  011102                  IJMP:   JMPCK               ;INDIRECT ADDRESS
4035
4036  011102  026727  000014  000006  JMPCK:  CMP     JMPSEQ,#6   ;CHECK JUMPS IN SEQUENCE: JMPSEQ
4037  011110  001405                          BEQ     TST136
4038                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4039                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4040                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4041                                                  ;           WHICH FOLLOWS W/ 627           <====
4042  011112  012742  000242                        MOV     #242,-(R2)  ;MOVE TO MAILBOX #  *******  242  *******
4043  011116  005242                                INC     -(R2)       ;SET MSGTYP TO FATAL ERROR
4044  011120  000000                                HALT                ;SHOULD ONLY BE HERE FROM MODE 6 JUMP
4045                                                                      ;  OR SEQUENCE ERROR
4046  011122  000000                  JMPSEQ: 0
```

```
4047
4048                    ;*****************************************************************************
4049                    ;
4050                    ;       THIS TEST VERIFIES ALL LEGAL MODES OF THE JSR INSTRUCTION.
4051                    ;THE CONCEPT OF LEAP FROGGING AND SEQUENCE CHECKING (JSRSEQ) IS
4052                    ;IDENTICAL TO THAT USED IN JMP TEST (SEE PREVIOUS TEST).  EACH
4053                    ;BLOCK OF CODE VERIFIES THE PREVIOUS JSR BY CHECKING THE SEQUENCE,
4054                    ;CHECKING THAT THE PC WAS SAVED IN THE SPECIFIED REGISTER, CHECKING
4055                    ;THAT THE SP WAS DECREMENTED, CHECKING THAT THE REGISTER WAS
4056                    ;SAVED ON THE STACK, AND FINALLY CHECKING THAT ANY MODE ADDRESS
4057                    ;REGISTER ALTERATIONS (E.G. INCREMENT REGISTER IN MODE 2) WERE
4058                    ;SUCCESSFUL.  R1 IS USED AS THE REGISTER IN ALL JSR INSTRUCTIONS.
4059                    ;       IF A FAILURE OCCURS, THE SEQUENCE CHECKER WILL ASSIST IN
4060                    ;DETERMINING JUST WHICH MODE FAILED.  IF THE SEQUENCE IS CORRECT
4061                    ;THEN THE ERROR DETECTED WAS A FUNCTIONAL FAILURE (E.G., INCORRECT
4062                    ;REGISTER SAVED).
4063                    ;
4064                    ;*****************************************************************************
4065                    ;TEST 136       TEST JSR INSTRUCTION W/ ALL MODES
4066                    ;*****************************************************************************
4067  011124  005212    TST136: INC     (R2)            ;UPDATE TEST NUMBER
4068  011126  022712  000136          CMP     #136,(R2)       ;SEQUENCE ERROR?
4069  011132  001001                  BNE     JSR0            ;BR TO ERROR HALT ON SEQ ERROR
4070  011134  000402                  BR      JSR1
4071  011136  000137  011572  JSR0:   JMP     @#JSRCK1
4072
4073  011142  012706  000500  JSR1:   MOV     #STBOT,R6       ;SET STACK POINTER
4074  011146  012700  011254          MOV     #JSR2,R0        ;SET TARGET ADDRESS
4075  011152  005037  011552          CLR     @#JSRSEQ        ;INITIALIZE SEQUENCE CHECKER
4076  011156  005001                  CLR     R1              ;INITIALIZE R1
4077  011160  005101                  COM     R1
4078  011162  004110                  JSR     R1,(R0)         ;TRY JSR MODE 1
4079                                                          ; TO SCOPE: REPLACE THE MOVE INSTRUCTION   <====
4080                                                          ;           FOLLOWING W/ 774               <====
4081  011164          JSR1A:
4082  011164  012742  000243          MOV     #243,-(R2)      ;MOVE TO MAILBOX #  *******  243  *******
4083  011170  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4084  011172  000000                  HALT                    ;JSR MODE 1 FAILED
4085
4086  011174  022737  000001  011552  JSR3:   CMP     #1,@# JSRSEQ    ;CHECK SEQUENCE: JSRSEQ=1?
4087  011202  001014                  BNE     JSR3A           ;BRANCH IF OUT OF SEQUENCE
4088  011204  020127  011336          CMP     R1,#JSR4        ;PROPER PC SAVED?
4089  011210  001011                  BNE     JSR3A           ;BRANCH IF PC WRONG
4090  011212  022706  000476          CMP     #STBOT-2,R6     ;STACK POINTER DECREMENTED?
4091  011216  001006                  BNE     JSR3A           ;BRANCH IF SP WRONG
4092  011220  022716  125252          CMP     #125252,(R6)    ;REG SAVED ON STACK?
4093  011224  001003                  BNE     JSR3A           ;BRANCH IF REG. NOT SAVED
4094  011226  022700  011176          CMP     #JSR3+2,R0      ;MODE 2 INCREMENT CORRECT?
4095  011232  001404                  BEQ     JSR3B
4096                                                          ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4097                                                          ;           CONDITIONAL BRANCH INST. AND   <====
4098                                                          ;           REPLACE THE MOVE INSTRUCTION   <====
4099                                                          ;           WHICH FOLLOWS W/ 740           <====
4100  011234          JSR3A:
4101  011234  012742  000244          MOV     #244,-(R2)      ;MOVE TO MAILBOX #  *******  244  *******
4102  011240  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
```

```
4103  011242  000000                        HALT                    ;JSR MODE 3 MALFUNCTIONED
4104  011244  005237  011552        JSR3B:  INC     @#JSRSEQ        ;UPDATE SEQUENCE CHECKER
4105  011250  004137  011336                JSR     R1,@#JSR4       ;TRY JSR MODE 4
4106
4107  011254  005737  011552        JSR2:   TST     @#JSRSEQ        ;CHECK SEQUENCE: JSRSEQ=0?
4108  011260  001011                        BNE     JSR2A           ;BRANCH IF OUT OF SEQUENCE
4109  011262  020127  011164                CMP     R1,#JSR1A       ;PROPER PC SAVED?
4110  011266  001006                        BNE     JSR2A           ;BRANCH IF PC WRONG
4111  011270  022706  000476                CMP     #STBOT-2,R6     ;R6 DECREMENT?
4112  011274  001003                        BNE     JSR2A           ;BRANCH IF R6 IS INCORRECT
4113  011276  021627  177777                CMP     (R6),#-1        ;REGISTER SAVED?
4114  011302  001404                        BEQ     JSR2B
4115                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4116                                                                ;           CONDITIONAL BRANCH INST. AND   <====
4117                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
4118                                                                ;           WHICH FOLLOWS W/ 714           <====
4119  011304                        JSR2A:
4120  011304  012742  000245                MOV     #245,-(R2)      ;MOVE TO MAILBOX #  *******  245  *******
4121  011310  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4122  011312  000000                        HALT                    ;JSR MODE 1 MALFUNCTIONED
4123  011314  012706  000500        JSR2B:  MOV     #STBOT,R6       ;INITIALIZE R6
4124  011320  012701  125252                MOV     #125252,R1      ;INITIALIZE R1
4125  011324  005237  011552                INC     @#JSRSEQ        ;UPDATE SEQUENCE CHECKER
4126  011330  012700  011174                MOV     #JSR3,R0        ;SET TARGET ADDRESS
4127  011334  004120                        JSR     R1,(R0)+        ;TRY JSR MODE 2
.28
4129  011336  022737  000002  011552  JSR4:   CMP     #2,@#JSRSEQ     ;CHECK SEQUENCE: JSRSEQ=2?
4130  011344  001003                        BNE     JSR4A           ;BRANCH IF OUT OF SEQUENCE
4131  011346  022701  011254                CMP     #JSR2,R1        ;PROPER PC SAVED?
4132  011352  001404                        BEQ     JSR4B
4133                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4134                                                                ;           CONDITIONAL BRANCH INST. AND   <====
4135                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
4136                                                                ;           WHICH FOLLOWS W/ 670           <====
4137  011354                        JSR4A:
4138  011354  012742  000246                MOV     #246,-(R2)      ;MOVE TO MAILBOX #  *******  246  *******
4139  011360  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4140  011362  000000                        HALT                    ;JSR MODE 3 MALFUNCTIONED
4141  011364  005237  011552        JSR4B:  INC     @#JSRSEQ        ;UPDATE SEQUENCE CHECKER
4142  011370  012700  011444                MOV     #JSR5+2,R0      ;SET TARGET ADDRESS
4143  011374  004140                        JSR     R1,-(R0)        ;TRY JSR MODE 4
4144
4145  011376  022767  000004  000146  JSR6:   CMP     #4,JSRSEQ       ;CHECK SEQUENCE: JSRSEQ=4?
4146  011404  001006                        BNE     JSR6A           ;BRANCH IF OUT OF SEQUENCE
4147  011406  022701  011510                CMP     #JSR7,R1        ;PROPER PC SAVED?
4148  011412  001003                        BNE     JSR6A           ;BRANCH IF PC WRONG
4149  011414  022700  011546                CMP     #JSR6AD,R0      ;MODE 5 REGISTER CORRECT?
4150  011420  001404                        BEQ     JSR6B
4151                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4152                                                                ;           CONDITIONAL BRANCH INST. AND   <====
4153                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
4154                                                                ;           WHICH FOLLOWS W/ 645           <====
4155  011422                        JSR6A:
4156  011422  012742  000247                MOV     #247,-(R2)      ;MOVE TO MAILBOX #  *******  247  *******
4157  011426  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4158  011430  000000                        HALT                    ;JSR MODE 5 FAILED
```

```
4159  011432  005237  011552          JSR6B:  INC     @#JSRSEQ        ;UPDATE SEQUENCE CHECKER
4160  011436  004167  000046                  JSR     R1,JSR7         ;TRY JSR MODE 6
4161  011442  022767  000003  000122  JSR5:   CMP     #3,JSRSEQ       ;CHECK SEQUENCE: JSRSEQ=3?
4162  011450  001006                          BNE     JSR5A           ;BRANCH IF OUT OF SEQUENCE
4163  011452  022701  011376                  CMP     #JSR6,R1        ;PROPER PC SAVED?
4164  011456  001003                          BNE     JSR5A           ;BRANCH IF PC WRONG
4165  011460  022700  011442                  CMP     #JSR5,R0        ;CHECK MODE 4 REGISTER
4166  011464  001404                          BEQ     JSR5B
4167                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4168                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4169                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4170                                                                  ;           WHICH FOLLOWS W/ 623           <====
4171  011466                          JSR5A:
4172  011466  012742  000250                  MOV     #250,-(R2)      ;MOVE TO MAILBOX #  *******  250  *******
4173  011472  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4174  011474  000000                          HALT                    ;JSR MODE 4 MALFUNCTIONED
4175  011476  005237  011552          JSR5B:  INC     @#JSRSEQ        ;UPDATE SEQUENCE CHECKER
4176  011502  012700  011550                  MOV     #JSR6AD+2,R0    ;POINT R0 TO TARGET ADDRESS
4177  011506  004150                          JSR     R1,@-(R0)       ;TRY JSR MODE 5
4178
4179  011510  022737  000005  011552  JSR7:   CMP     #5,@#JSRSEQ     ;CHECK SEQUENCE: JSRSEQ=5?
4180  011516  001003                          BNE     JSR7A           ;BRANCH IF OUT OF SEQUENCE
4181  011520  022701  011442                  CMP     #JSR5,R1        ;PROPER PC SAVED?
4182  011524  001404                          BEQ     JSR7B
4183                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4184                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4185                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4186                                                                  ;           WHICH FOLLOWS W/ 603           <====
4187  011526                          JSR7A:
4188  011526  012742  000251                  MOV     #251,-(R2)      ;MOVE TO MAILBOX #  *******  251  *******
4189  011532  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4190  011534  000000                          HALT                    ;JSR MODE 6 FAILED
4191  011536  005237  011552          JSR7B:  INC     @#JSRSEQ        ;UPDATE SEQUENCE CHECKER
4192  011542  004177  000002                  JSR     R1,@JSRCKAD     ;TRY JSR MODE 7
4193
4194  011546  011376          JSR6AD: JSR6                            ;MODE 5 TARGET ADDRESS
4195  011550  011554          JSRCKAD:JSRCK                           ;MODE 7 TARGET ADDRESS
4196  011552  000000          JSRSEQ: 0                               ;SEQUENCE CHECKER
4197
4198  011554  022767  000006  177770  JSRCK:  CMP     #6,JSRSEQ       ;CHECK SEQUENCE: JSRSEQ=6?
4199  011562  001003                          BNE     JSRCK1          ;BRANCH IF OUT OF SEQUENCE
4200  011564  022701  011546                  CMP     #JSR6AD,R1      ;PROPER PC SAVED?
4201  011570  001404                          BEQ     TST137
4202                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4203                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4204                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4205                                                                  ;           WHICH FOLLOWS W/ 561           <====
4206  011572                          JSRCK1:
4207  011572  012742  000252                  MOV     #252,-(R2)      ;MOVE TO MAILBOX #  *******  252  *******
4208  011576  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4209  011600  000000                          HALT                    ;JSR MODE 7 MALFUNCTIONED
4210                                                                  ;  OR SEQUENCE ERROR
4211
4212
4213                          ;*****************************************************************************
4214                          ;
```

```
                                ;       THIS TEST VERIFIES THE RTS INSTRUCTION.  THE STACK POINTER
                                ;IS INITIALIZED AND A TEST PATTERN STORED ON STACK.  R0 IS LOADED
                                ;WITH RETURN ADDRESS.  AN RTS IS EXECUTED, AND, AT THE TARGET
                                ;ADDRESS, A CHECK IS MADE THAT R0 WAS PROPERLY RESTORED FROM THE
                                ;STACK.
                                ;*****************************************************************************
                                ;TEST 137       TEST RTS INSTRUCTION
                                ;*****************************************************************************
011602  005212                TST137: INC     (R2)            ;UPDATE TEST NUMBER
011604  022712  000137              CMP     #137,(R2)       ;SEQUENCE ERROR?
011610  001016                      BNE     TST140-10       ;BR TO ERROR HALT ON SEQ ERROR
011612  012706  000500              MOV     #STBOT,R6       ;INITIALIZE STACK POINTER
011616  012746  052525              MOV     #52525,-(R6)    ;INITIALIZE TOP OF STACK
011622  012700  011640              MOV     #RTS1,R0        ;INITIALIZE RETURN REGISTER
011626  000200                      RTS     R0              ;TRY RTS THROUGH R0
                                                            ; TO SCOPE: REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           FOLLOWING W/ 770               <====
011630  012742  000253              MOV     #253,-(R2)      ;MOVE TO MAILBOX #  *******  253  *******
011634  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
011636  000000                      HALT                    ;RTS FAILED
011640  022700  052525      RTS1:   CMP     #52525,R0       ;CHECK THAT R0 RESTORED FROM STACK
011644  001404                      BEQ     TST140
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 762           <====
011646  012742  000254              MOV     #254,-(R2)      ;MOVE TO MAILBOX #  *******  254  *******
011652  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
011654  000000                      HALT                    ;RTS MALFUNCTIONED
                                                            ;  OR SEQUENCE ERROR
```

```
                                     ;*******************************************************************************
                                     ;
                                     ;        THESE NEXT FOUR TESTS VERIFY THE FUNCTIONING OF A GROUP
                                     ;OF FOUR INSTRUCTIONS.  THE GROUP CONSISTS OF THE INSTRUCTIONS:
                                     ;MOV, BIC, BIT, AND BIS.  THESE INSTRUCTIONS ARE SIMILAR IN THE
                                     ;WAY THEY EFFECT THE C AND V BITS.  THEY ALL LEAVE THE V-BIT
                                     ;CLEAR AND THE C-BIT UNAFFECTED.
                                     ;        THE TEST PROCEDURE IS AS FOLLOWS:  THE N, Z, AND V BITS
                                     ;ARE LOADED WITH THE COMPLEMENT OF THE EXPECTED RESULTS, THE C-BIT
                                     ;IS LOADED WITH THE DESIRED RESULT.  THE INSTRUCTION IS EXECUTED
                                     ;WITH DIFFERENT DATA PATTERNS AND THE RESULTS ARE VERIFIED WITH
                                     ;A SERIES OF CONDITIONAL BRANCH INSTRUCTIONS.  THE DATA IS CHOSEN
                                     ;TO PRODUCT ALL POSSIBLE COMBINATIONS OF THE C AND V BITS.
                                     ;
                                     ;*******************************************************************************
                                     ;TEST 140        TEST MOV INSTRUCTION
                                     ;*******************************************************************************
011656  005212                       TST140: INC     (R2)            ;UPDATE TEST NUMBER
011660  022712  000140                       CMP     #140,(R2)       ;SEQUENCE ERROR?
011664  001022                               BNE     TST141-10       ;BR TO ERROR HALT ON SEQ ERROR
011666  000277                               SCC                     ;CC=0110
011670  000251                               +CLN!CLC
011672  012700  100000                       MOV     #100000,R0      ;CC=1000
011676  101402                               BLOS    MOV1
011700  102401                               BVS     MOV1
011702  100404                               BMI     MOV2
                                                                     ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                     ;           CONDITIONAL BRANCH INST. AND    <====
                                                                     ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                     ;           WHICH FOLLOWS W/ 771            <====
011704                               MOV1:
011704  012742  000255                       MOV     #255,-(R2)      ;MOVE TO MAILBOX #  *******  255  *******
011710  005242                               INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
011712  000000                               HALT                    ;MOV DID NOT SET CC'S CORRECTLY

011714  000277                       MOV2:   SCC                     ;CC=1011
011716  000244                               CLZ
011720  012700  000000                       MOV     #0,R0           ;CC=0101
011724  101002                               BHI     MOV3            ;C OR Z = 0?
011726  102401                               BVS     MOV3            ;V=1?
011730  100004                               BPL     TST141
                                                                     ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                     ;           CONDITIONAL BRANCH INST. AND    <====
                                                                     ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                     ;           WHICH FOLLOWS W/ 756            <====
011732                               MOV3:
011732  012742  000256                       MOV     #256,-(R2)      ;MOVE TO MAILBOX #  *******  256  *******
011736  005242                               INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
011740  000000                               HALT                    ;MOV DID NOT SET CC'S CORRECTLY
                                                                     ;  OR SEQUENCE ERROR


                                     ;*******************************************************************************
                                     ;TEST 141        TEST BIT INSTRUCTION
                                     ;*******************************************************************************
011742  005212                       TST141: INC     (R2)            ;UPDATE TEST NUMBER
```

```
4301  011744  022712  000141                  CMP     #141,(R2)       ;SEQUENCE ERROR?
4302  011750  001024                          BNE     TST142-10       ;BR TO ERROR HALT ON SEQ ERROR
4303  011752  012700  100001                  MOV     #100001,R0
4304  011756  000277                          SCC                     ;CC=0110
4305  011760  000251                          +CLN!CLC
4306  011762  032700  100000                  BIT     #100000,R0      ;CC=1000
4307  011766  101402                          BLOS    BIT1
4308  011770  102401                          BVS     BIT1
4309  011772  100404                          BMI     BIT2
4310                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4311                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4312                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4313                                                                  ;           WHICH FOLLOWS W/ 767           <====
4314  011774                          BIT1:
4315  011774  012742  000257                  MOV     #257,-(R2)      ;MOVE TO MAILBOX #  *******  257  *******
4316  012000  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4317  012002  000000                          HALT                    ;BIT DID NOT SET CC'S CORRECTLY
4318
4319  012004  000277                  BIT2:   SCC                     ;CC=1011
4320  012006  000244                          CLZ
4321  012010  032700  077776                  BIT     #77776,R0       ;CC=0101
4322  012014  101002                          BHI     BIT3
4323  012016  102401                          BVS     BIT3
4324  012020  100004                          BPL     TST142
4325                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4326                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4327                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4328                                                                  ;           WHICH FOLLOWS W/ 754           <====
4329  012022                          BIT3:
4330  012022  012742  000260                  MOV     #260,-(R2)      ;MOVE TO MAILBOX #  *******  260  *******
4331  012026  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4332  012030  000000                          HALT                    ;BIT DID NOT SET CC'S CORRECTLY
4333                                                                  ;  OR SEQUENCE ERROR
4334
4335                                  ;*****************************************************************************
4336                                  ;TEST 142       TEST BIC INSTRUCTION
4337                                  ;*****************************************************************************
4338  012032  005212                  TST142: INC     (R2)            ;UPDATE TEST NUMBER
4339  012034  022712  000142                  CMP     #142,(R2)       ;SEQUENCE ERROR?
4340  012040  001024                          BNE     TST143-10       ;BR TO ERROR HALT ON SEQ ERROR
4341  012042  012700  177777                  MOV     #177777,R0
4342  012046  000277                          SCC                     ;CC=0110
4343  012050  000251                          +CLN!CLC
4344  012052  042700  077777                  BIC     #77777,R0       ;CC=1000
4345  012056  101402                          BLOS    BIC1
4346  012060  102401                          BVS     BIC1
4347  012062  100404                          BMI     BIC2
4348                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4349                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4350                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
4351                                                                  ;           WHICH FOLLOWS W/ 767           <====
4352  012064                          BIC1:
4353  012064  012742  000261                  MOV     #261,-(R2)      ;MOVE TO MAILBOX #  *******  261  *******
4354  012070  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4355  012072  000000                          HALT                    ;BIC DID NOT SET CC'S CORRECTLY
4356  012074  000277                  BIC2:   SCC                     ;CC=1011
```

```
4357  012076  000244                          CLZ
4358  012100  042700  100000                  BIC     #100000,R0      ;CC=0101
4359  012104  101002                          BHI     BIC3
4360  012106  102401                          BVS     BIC3
4361  012110  100004                          BPL     TST143
4362                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
4363                                                  ;           CONDITIONAL BRANCH INST. AND    <====
4364                                                  ;           REPLACE THE MOVE INSTRUCTION    <====
4365                                                  ;           WHICH FOLLOWS W/ 754            <====
4366  012112                          BIC3:
4367  012112  012742  000262                  MOV     #262,-(R2)      ;MOVE TO MAILBOX #  *******  262  *******
4368  012116  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4369  012120  000000                          HALT                    ;BIC DID NOT SET CC'S CORRECTLY
4370                                                                  ;  OR SEQUENCE ERROR
4371
4372                                  ;*****************************************************************************
4373                                  ;TEST 143       TEST BIS INSTRUCTION
4374                                  ;*****************************************************************************
4375  012122  005212                  TST143: INC     (R2)            ;UPDATE TEST NUMBER
4376  012124  022712  000143                  CMP     #143,(R2)       ;SEQUENCE ERROR?
4377  012130  001025                          BNE     TST144-10       ;BR TO ERROR HALT ON SEQ ERROR
4378  012132  005000                          CLR     R0              ;R0=0
4379  012134  000277                          SCC                     ;CC=1010
4380  012136  000251                          +CLN!CLC
4381  012140  052700  000000                  BIS     #0,R0           ;CC=0100  R0=0
4382  012144  103403                          BCS     BIS1
4383  012146  102402                          BVS     BIS1
4384  012150  100401                          BMI     BIS1
4385  012152  001404                          BEQ     BIS2
4386                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
4387                                                  ;           CONDITIONAL BRANCH INST. AND    <====
4388                                                  ;           REPLACE THE MOVE INSTRUCTION    <====
4389                                                  ;           WHICH FOLLOWS W/ 767            <====
4390  012154                          BIS1:
4391  012154  012742  000263                  MOV     #263,-(R2)      ;MOVE TO MAILBOX #  *******  263  *******
4392  012160  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4393  012162  000000                          HALT                    ;BIS DID NOT SET CC'S CORRECTLY
4394  012164  000277                  BIS2:   SCC                     ;CC=0111
4395  012166  000250                          CLN
4396  012170  052700  177777                  BIS     #177777,R0      ;CC=1001
4397  012174  103003                          BCC     BIS3
4398  012176  102402                          BVS     BIS3
4399  012200  001401                          BEQ     BIS3
4400  012202  100404                          BMI     TST144
4401                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
4402                                                  ;           CONDITIONAL BRANCH INST. AND    <====
4403                                                  ;           REPLACE THE MOVE INSTRUCTION    <====
4404                                                  ;           WHICH FOLLOWS W/ 753            <====
4405  012204                          BIS3:
4406  012204  012742  000264                  MOV     #264,-(R2)      ;MOVE TO MAILBOX #  *******  264  *******
4407  012210  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4408  012212  000000                          HALT                    ;BIS DID NOT SET CC'S CORRECTLY
4409                                                                  ;  OR SEQUENCE ERROR
4410
```

```
4411
4412                    ;*****************************************************************************
4413                    ;
4414                    ;       THESE NEXT TWO TESTS VERIFY THE FUNCTIONING OF THE INC AND
4415                    ;DEC INSTRUCTIONS.  THESE INSTRUCTIONS BOTH EFFECT THE C AND V
4416                    ;BITS THE SAME; THE C-BIT IS LEFT UNCHANGED AND THE V-BIT IS DEPENDENT
4417                    ;UPON THE DATA RESULTS.  THE SAME PROCEDURE IS USED.  THE CONDITION
4418                    ;CODE BITS ARE INITIALIZED, THE INSTRUCTION IS EXECUTED AND THE
4419                    ;RESULTS ARE VERIFIED WITH A SERIES OF CONDITIONAL BRANCH INSTRUCTIONS.
4420                    ;THIS PROCEDURE IS REPEATED WITH SEVERAL DATA PATTERNS TO PRODUCE
4421                    ;DIFFERENT COMBINATIONS OF THE C AND V BITS.
4422                    ;
4423                    ;*****************************************************************************
4424                    ;TEST 144       TEST INC INSTRUCTION
4425                    ;*****************************************************************************
4426  012214  005212    TST144: INC     (R2)            ;UPDATE TEST NUMBER
4427  012216  022712    000144  CMP     #144,(R2)       ;SEQUENCE ERROR?
4428  012222  001037            BNE     TST145-10       ;BR TO ERROR HALT ON SEQ ERROR
4429  012224  012700    077777  MOV     #077777,R0      ;R0=077777
4430  012230  000257            CCC                     ;CC=0100
4431  012232  000264            SEZ
4432  012234  005200            INC     R0              ;CC=1010   R0=10000
4433  012236  101402            BLOS    INC1
4434  012240  100001            BPL     INC1
4435  012242  102404            BVS     INC2
4436                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4437                                                    ;           CONDITIONAL BRANCH INST. AND   <====
4438                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
4439                                                    ;           WHICH FOLLOWS W/ 770           <====
4440  012244            INC1:
4441  012244  012742    000265          MOV     #265,-(R2)      ;MOVE TO MAILBOX #  *******  265  *******
4442  012250  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4443  012252  000000                    HALT                    ;INC DID NOT SET CC'S CORRECTLY
4444  012254  052700    077777  INC2:   BIS     #77777,R0       ;R0=177777
4445  012260  000261            SEC                     ;CC=1011
4446  012262  000244            CLZ
4447  012264  005200            INC     R0              ;CC=0101   R0=0
4448  012266  100403            BMI     INC3
4449  012270  102402            BVS     INC3
4450  012272  103001            BCC     INC3
4451  012274  001404            BEQ     INC4
4452                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4453                                                    ;           CONDITIONAL BRANCH INST. AND   <====
4454                                                    ;           REPLACE THE MOVE INSTRUCTION   <====
4455                                                    ;           WHICH FOLLOWS W/ 753           <====
4456  012276            INC3:
4457  012276  012742    000266          MOV     #266,-(R2)      ;MOVE TO MAILBOX #  *******  266  *******
4458  012302  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4459  012304  000000                    HALT                    ;INC DID NOT SET CC'S CORRECTLY
4460
4461  012306  000277    INC4:   SCC                     ;CC=1110
4462  012310  000241            CLC
4463  012312  005200            INC     R0              ;CC=0000   R0=1
4464  012314  101402            BLOS    INC5
4465  012316  100401            BMI     INC5
4466  012320  100004            BPL     TST145
```

```
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 741           <====
012322                          INC5:
012322  012742  000267                MOV     #267,-(R2)      ;MOVE TO MAILBOX #  *******  267  *******
012326  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
012330  000000                        HALT                    ;INC DID NOT SET CC'S CORRECTLY
                                                                ;  OR SEQUENCE ERROR

                                ;*****************************************************************************
                                ;TEST 145       TEST DEC INSTRUCTION
                                ;*****************************************************************************
012332  005212                TST145: INC     (R2)            ;UPDATE TEST NUMBER
012334  022712  000145                CMP     #145,(R2)       ;SEQUENCE ERROR?
012340  001051                        BNE     TST146-10       ;BR TO ERROR HALT ON SEQ ERROR
012342  012700  000002                MOV     #2,R0           ;R0=2
012346  000277                        SCC                     ;CC=1111
012350  005300                        DEC     R0              ;CC=0001   R0=1
012352  100403                        BMI     DEC1
012354  001402                        BEQ     DEC1
012356  102401                        BVS     DEC1
012360  103404                        BCS     DEC2
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 770           <====
012362                          DEC1:
012362  012742  000270                MOV     #270,-(R2)      ;MOVE TO MAILBOX #  *******  270  *******
012366  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
012370  000000                        HALT                    ;DEC DID NOT SET CC'S CORRECTLY
012372  000261                DEC2:   SEC                     ;CC=1011
012374  000244                        CLZ
012376  005300                        DEC     R0              ;CC=0101   R0=0
012400  101002                        BHI     DEC3
012402  100401                        BMI     DEC3
012404  102004                        BVC     DEC4
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 756           <====
012406                          DEC3:
012406  012742  000271                MOV     #271,-(R2)      ;MOVE TO MAILBOX #  *******  271  *******
012412  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
012414  000000                        HALT                    ;DEC DID NOT SET CC'S CORRECTLY
012416  000277                DEC4:   SCC                     ;CC=0110
012420  000251                        +CLN!CLC
012422  005300                        DEC     R0              ;CC=1000   R0=177777
012424  101402                        BLOS    DEC5
012426  102401                        BVS     DEC5
012430  100404                        BMI     DEC6
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 744           <====
012432                          DEC5:
```

```
4523  012432  012742  000272                  MOV     #272,-(R2)      ;MOVE TO MAILBOX #  *******  272  *******
4524  012436  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4525  012440  000000                          HALT                    ;DEC DID NOT SET CC'S CORRECTLY
4526  012442  042700  077777          DEC6:   BIC     #77777,R0       ;R0=100000
4527  012446  000277                          SCC                     ;CC=0101
4528  012450  000252                          +CLN!CLV
4529  012452  005300                          DEC     R0              ;CC=1011   R0=77777
4530  012454  100403                          BMI     DEC7            ;CC=0011
4531  012456  001402                          BEQ     DEC7
4532  012460  102001                          BVC     DEC7
4533  012462  103404                          BCS     TST146
4534                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4535                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
4536                                                                  ;           REPLACE THE MOVE INSTRUCTION    ====
4537                                                                  ;           WHICH FOLLOWS W/ 727           <====
4538  012464                          DEC7:
4539  012464  012742  000273                  MOV     #273,-(R2)      ;MOVE TO MAILBOX #  *******  273  *******
4540  012470  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4541  012472  000000                          HALT                    ;DEC DID NOT SET CC'S CORRECTLY
4542                                                                  ;  OR SEQUENCE ERROR
4543
```

```
                                ;*****************************************************************************
                                ;
                                ;       THESE NEXT THREE TESTS VERIFY THE FUNCTIONING OF THE CLR,
                                ;TST, AND SWAB INSTRUCTIONS.  THESE THREE INSTRUCTIONS ALL LEAVE
                                ;THE C AND V BITS CLEARED.  AGAIN, THE CONDITION CODES ARE PRESET,
                                ;THE INSTRUCTION EXECUTED AND THE RESULTS CHECKED WITH CONDITIONAL
                                ;BRANCH INSTRUCTIONS.  THE PROCEDURE IS REPEATED TO PRODUCE OTHER
                                ;COMBINATIONS OF CONDITION CODES.
                                ;
                                ;*****************************************************************************
                                ;TEST 146       TEST CLR INSTRUCTION
                                ;*****************************************************************************
012474  005212                TST146: INC     (R2)            ;UPDATE TEST NUMBER
012476  022712  000146              CMP     #146,(R2)       ;SEQUENCE ERROR?
012502  001007                      BNE     TST147-10       ;BR TO ERROR HALT ON SEQ ERROR
012504  000277                      SCC                     ;CC=1011
012506  000244                      CLZ
012510  005000                      CLR     R0              ;CC=0100   R0=0
012512  100403                      BMI     CLR1
012514  102402                      BVS     CLR1
012516  103401                      BCS     CLR1
012520  001404                      BEQ     TST147
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 771           <====
012522                          CLR1.
012522  012742  000274                MOV     #274,-(R2)      ;MOVE TO MAILBOX #  *******  274  *******
012526  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
012530  000000                      HALT                    ;CLR DID NOT SET CC'S CORRECTLY
                                                            ;  OR SEQUENCE ERROR
                                ;*****************************************************************************
                                ;TEST 147       TEST TST INSTRUCTION
                                ;*****************************************************************************
012532  005212                TST147: INC     (R2)            ;UPDATE TEST NUMBER
012534  022712  000147              CMP     #147,(R2)       ;SEQUENCE ERROR?
012540  001022                      BNE     TST150-10       ;BR TO ERROR HALT ON SEQ ERROR
012542  000277                      SCC                     ;CC=1011
012544  000244                      CLZ
012546  005700                      TST     R0              ;CC=0100
012550  100403                      BMI     TEST1
012552  102402                      BVS     TEST1
012554  103401                      BCS     TEST1
012556  001404                      BEQ     TEST2
                                                            ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                            ;           CONDITIONAL BRANCH INST. AND   <====
                                                            ;           REPLACE THE MOVE INSTRUCTION   <====
                                                            ;           WHICH FOLLOWS W/ 771           <====
012560                          TEST1:
012560  012742  000275                MOV     #275,-(R2)      ;MOVE TO MAILBOX #  *******  275  *******
012564  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
012566  000000                      HALT                    ;TEST DID NOT SET CC'S CORRECTLY
012570  005300                TEST2:  DEC     R0              ;MAKE R0 NEGATIVE
012572  000277                      SCC                     ;CC=0111
```

```
4600  012574  000250                    CLN
4601  012576  005700                    TST     R0              ;CC=1000
4602  012600  101402                    BLOS    TEST3
4603  012602  102401                    BVS     TEST3
4604  012604  100404                    BMI     TST15C
                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                        ;           WHICH FOLLOWS W/ 756           <====
4609  012606                    TEST3:
4610  012606  012742  000276            MOV     #276,-(R2)      ;MOVE TO MAILBOX #  *******  276  *******
4611  012612  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4612  012614  000000                    HALT                    ;TEST DID NOT SET CC'S CORRECTLY
                                        ;  OR SEQUENCE ERROR
4615  ;*****************************************************************************
4616  ;TEST 150       TEST SWAB INSTRUCTION
4617  ;*****************************************************************************
4618  012616  005212            TST150: INC     (R2)            ;UPDATE TEST NUMBER
4619  012620  022712  000150            CMP     #150,(R2)       ;SEQUENCE ERROR?
4620  012624  001023                    BNE     TST151-10       ;BR TO ERROR HALT ON SEQ ERROR
4621  012626  012700  170000            MOV     #170000,R0      ;R0=170000
4622  012632  000277                    SCC                     ;CC=0111
4623  012634  000250                    CLN
4624  012636  000300                    SWAB    R0              ;CC=1000   R0=360
4625  012640  101402                    BLOS    SWB1
4626  012642  102401                    BVS     SWB1
4627  012644  100404                    BMI     SWB2
                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                        ;           WHICH FOLLOWS W/ 77C           <====
4632  012646                    SWB1:
4633  012646  012742  000277            MOV     #277,-(R2)      ;MOVE TO MAILBOX #  *******  277  *******
4634  012652  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4635  012654  000000                    HALT                    ;SWAB DID NOT SET CC'S CORRECTLY
4636  012656  000277            SWB2:   SCC                     ;CC=1011
4637  012660  000244                    CLZ
4638  012662  000300                    SWAB    R0              ;CC=0100   R0=170000
4639  012664  102403                    BVS     SWB3
4640  012666  103402                    BCS     SWB3
4641  012670  100401                    BMI     SWB3
4642  012672  001404                    BEQ     TST151
                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                        ;           WHICH FOLLOWS W/ 755           <====
4647  012674                    SWB3:
4648  012674  012742  000300            MOV     #300,-(R2)      ;MOVE TO MAILBOX #  *******  300  *******
4649  012700  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4650  012702  000000                    HALT                    ;
4651
```

```
                                ;*****************************************************************************
                                ;
                                ;       THESE NEXT TWO TESTS VERIFY THE FUNCTIONING OF THE ADD AND
                                ;ADC INSTRUCTIONS.  BOTH OF THESE INSTRUCTIONS HANDLE THE C AND
                                ;V BITS IDENTICALLY.  THE PROCEDURE IS TO PRESET THE CONDITION
                                ;CODES, EXECUTE THE INSTRUCTION WITH A PARTICULAR SET OF DATA, AND
                                ;THEN CHECK THE RESULTS BY EXECUTING A SERIES OF CONDITIONAL
                                ;BRANCHES.  THES PROCEDURE IS REPEATED SEVERAL TIMES WITH DIFFERENT
                                ;DATA TO PRODUCE EVERY COMBINATION OF C AND V BITS.
                                ;
                                ;*****************************************************************************
                                ;TEST 151        TEST ADD INSTRUCTION
                                ;*****************************************************************************
012704  005212                TST151: INC     (R2)            ;UPDATE TEST NUMBER
012706  022712  000151              CMP     #151,(R2)       ;SEQUENCE ERROR?
012712  001062                      BNE     TST152-10       ;BR TO ERROR HALT ON SEQ ERROR
012714  012700  040000              MOV     #40000,R0       ;R0=40000
012720  000277                      SCC                     ;CC=1111
012722  062700  030000              ADD     #30000,R0       ;CC=0000   R0=70000
012726  101402                      BLOS    ADD1
012730  102401                      BVS     ADD1
012732  100004                      BPL     ADD2
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 770           <====
012734                  ADD1:
012734  012742  000301              MOV     #301,-(R2)      ;MOVE TO MAILBOX #  *******  301  *******
012740  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
012742  000000                      HALT                    ;ADD DID NOT SET CC'S CORRECTLY
012744  000264                ADD2: SEZ                     ;CC=0100

012746  062700  010000              ADD     #10000,R0       ;CC=1010   40=100000
012752  101402                      BLOS    ADD3
012754  102001                      BVC     ADD3
012756  100404                      BMI     ADD4
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 756           <====
012760                  ADD3:
012760  012742  000302              MOV     #302,-(R2)      ;MOVE TO MAILBOX #  *******  302  *******
012764  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
012766  000000                      HALT                    ;ADD DID NOT SET CC'S CORRECTLY
012770  000257                ADD4: CCC                     ;CC=1000
012772  000270                      SEN
012774  062700  100000              ADD     #100000,R0      ;CC=0111   R0=0
013000  101002                      BHI     ADD5
013002  102001                      BVC     ADD5
013004  100004                      BPL     ADD6
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 743           <====
013006                  ADD5:
```

```
4708  013006  012742  000303              MOV     #303,-(R2)      ;MOVE TO MAILBOX #  *******  303  *******
4709  013012  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4710  013014  000000                      HALT                    ;ADD DID NOT SET CC'S CORRECTLY
4711  013016  062700  177777    ADD6:     ADD     #177777,R0      ;CC=1000   R0=177777
4712  013022  101402                      BLOS    ADD7
4713  013024  102401                      BVS     ADD7
4714  013026  100404                      BMI     ADD8
4715                                                              ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4716                                                              ;           CONDITIONAL BRANCH INST. AND   <====
4717                                                              ;           REPLACE THE MOVE INSTRUCTION   <====
4718                                                              ;           WHICH FOLLOWS W/ 732           <====
4719  013030            ADD7:
4720  013030  012742  000304              MOV     #304,-(R2)      ;MOVE TO MAILBOX #  *******  304  *******
4721  013034  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4722  013036  000000                      HALT                    ;ADD DID NOT SET CC'S CORRECTLY
4723  013040  000277    ADD8:     SCC                             ;CC=1010
4724  013042  000245                      +CLC!CLZ
4725  013044  062700  000001              ADD     #1,R0           ;CC=0101   R=0
4726  013050  102403                      BVS     ADD9
4727  013052  103002                      BCC     ADD9
4728  013054  100401                      BMI     ADD9
4729  013056  001404                      BEQ     TST152
4730                                                              ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4731                                                              ;           CONDITIONAL BRANCH INST. AND   <====
4732                                                              ;           REPLACE THE MOVE INSTRUCTION   <====
4733                                                              ;           WHICH FOLLOWS W/ 716           <====
4734  013060            ADD9:
4735  013060  012742  000305              MOV     #305,-(R2)      ;MOVE TO MAILBOX #  *******  305  *******
4736  013064  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4737  013066  000000                      HALT                    ;ADD DID NOT SET CC'S CORRECTLY
4738                                                              ;  OR SEQUENCE ERROR
4739
4740                      ;*****************************************************************************
4741                      ;TEST 152       TEST ADC INSTRUCTION
4742                      ;*****************************************************************************
4743  013070  005212    TST152:   INC     (R2)            ;UPDATE TEST NUMBER
4744  013072  022712  000152              CMP     #152,(R2)       ;SEQUENCE ERROR?
4745  013076  001037                      BNE     TST153-10       ;BR TO ERROR HALT ON SEQ ERROR
4746  013100  012700  077777              MOV     #077777,R0
4747  013104  000277                      SCC                     ;CC=0101
4748  013106  000252                      +CLN!CLV
4749  013110  005500                      ADC     R0              ;CC=1010
4750  013112  101402                      BLOS    ADC1
4751  013114  102001                      BVC     ADC1
4752  013116  100404                      BMI     ADC2
4753                                                              ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
4754                                                              ;           CONDITIONAL BRANCH INST. AND   <====
4755                                                              ;           REPLACE THE MOVE INSTRUCTION   <====
4756                                                              ;           WHICH FOLLOWS W/ 770           <====
4757  013120            ADC1:
4758  013120  012742  000306              MOV     #306,-(R2)      ;MOVE TO MAILBOX #  *******  306  *******
4759  013124  005242                      INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4760  013126  000000                      HALT                    ;ADC DID NOT SET CC'S CORRECTLY
4761  013130  052700  077777    ADC2:     BIS     #77777,R0
4762  013134  000277                      SCC                     ;CC=1011
4763  013136  000244                      CLZ
```

```
 4764  013140  005500                        ADC     R0              ;CC=0101   R0=0
 4765  013142  101002                        BHI     ADC3
 4766  013144  102401                        BVS     ADC3
 4767  013146  100004                        BPL     ADC4
 4768                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 4769                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
 4770                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
 4771                                                                  ;           WHICH FOLLOWS W/ 754           <====
 4772  013150                          ADC3:
 4773  013150  012742  000307                MOV     #307,-(R2)      ;MOVE TO MAILBOX #  *******  307  *******
 4774  013154  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 4775  013156  000000                        HALT                    ;ADC DID NOT SET CC'S CORRECTLY
 4776  013160  000277                ADC4:   SCC
 4777  013162  000245                        +CLZ!CLC                        ;CC=1010
 4778  013164  005500                        ADC     R0              ;CC=0100
 4779  013166  102403                        BVS     ADC5
 4780  013170  103402                        BCS     ADC5
 4781  013172  100401                        BMI     ADC5
 4782  013174  001404                        BEQ     TST153
 4783                                                                  ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 4784                                                                  ;           CONDITIONAL BRANCH INST. AND   <====
 4785                                                                  ;           REPLACE THE MOVE INSTRUCTION   <====
 4786                                                                  ;           WHICH FOLLOWS W/ 741           <====
 4787  013176                          ADC5:
 4788  013176  012742  000310                MOV     #310,-(R2)      ;MOVE TO MAILBOX #  *******  310  *******
 4789  013202  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 4790  013204  000000                        HALT                    ;ADC DID NOT SET CC'S CORRECTLY
 4791                                                                  ;  OR SEQUENCE ERROR
```

```
                                ;*****************************************************************************
                                ;
                                ;       THESE NEXT THREE TESTS VERIFY THE FUNCTIONING OF THE NEG,
                                ;CMP, AND COM INSTRUCTIONS.  EACH OF THESE INSTRUCTIONS GENERATE
                                ;THE C AND V BITS IDENTICALLY.  THE CONDITION CODES ARE PRESET,
                                ;THE INSTRUCTIONS EXECUTED, AND THE RESULTS CHECKED WITH A SERIES
                                ;OF CONDITIONAL BRANCH INSTRUCTIONS.  THIS PROCEDURE IS REPEATED
                                ;SEVERAL TIMES WITH DIFFERENT DATA IN ORDER TO GENERATE DIFFERENT
                                ;COMBINATIONS OF THE C AND V BITS.
                                ;
                                ;*****************************************************************************
                                ;TEST 153        TEST NEG INSTRUCTION
                                ;*****************************************************************************
013206  005212                  TST153: INC     (R2)            ;UPDATE TEST NUMBER
013210  022712  000153                  CMP     #153,(R2)       ;SEQUENCE ERROR?
013214  001042                          BNE     TST154-10       ;BR TO ERROR HALT ON SEQ ERROR
013216  012700  000001                  MOV     #1,R0
013222  000277                          SCC                     ;CC=0110
013224  000251                          +CLN!CLC
013226  005400                          NEG     R0              ;CC=1001   R0=177777
013230  103003                          BCC     NEG1
013232  102402                          BVS     NEG1
013234  001401                          BEQ     NEG1
013236  100404                          BMI     NEG2
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 767           <====
013240                          NEG1:
013240  012742  000311                  MOV     #311,-(R2)      ;MOVE TO MAILBOX #  *******  311  *******
013244  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
013246  000000                          HALT                    ;NEG DID NOT SET CC'S CORRECTLY
013250  042700  077777          NEG2:   BIC     #77777,R0
013254  000257                          CCC                     ;CC=0100
013256  000264                          SEZ
013260  005400                          NEG     R0              ;CC=1011   R0=100000
013262  102003                          BVC     NEG3
013264  103002                          BCC     NEG3
013266  001401                          BEQ     NEG3
013270  100404                          BMI     NEG4
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                                ;           CONDITIONAL BRANCH INST. AND   <====
                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
                                                                ;           WHICH FOLLOWS W/ 752           <====
013272                          NEG3:
013272  012742  000312                  MOV     #312,-(R2)      ;MOVE TO MAILBOX #  *******  312  *******
013276  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
013300  000000                          HALT                    ;NEG DID NOT SET CC'S CORRECTLY
013302  005000          NEG4:           CLR     R0
013304  000277                          SCC                     ;CC=1011
013306  000244                          CLZ
013310  005400                          NEG     R0              ;CC=0100   R0=0
013312  102403                          BVS     NEG5
013314  103402                          BCS     NEG5
013316  001001                          BNE     NEG5
```

```
013320 100004                          BPL     TST154
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 736           <====
013322                          NEG5:
013322 012742 000313                   MOV     #313,-(R2)     ;MOVE TO MAILBOX #  *******  313  *******
013326 005242                          INC     -(R2)          ;SET MSGTYP TO FATAL ERROR
013330 000000                          HALT                   ;NEG DID NOT SET CC'S CORRECTLY
                                                        ;  OR SEQUENCE ERROR

                                ;*************************************************************************
                                ;TEST 154       TEST CMP INSTRUCTION
                                ;*************************************************************************
013332 005212                   TST154: INC     (R2)           ;UPDATE TEST NUMBER
013334 022712 000154                   CMP     #154,(R2)      ;SEQUENCE ERROR?
013340 001060                          BNE     TST155-10      ;BR TO ERROR HALT ON SEQ ERROR
013342 012700 000005                   MOV     #5,R0
013346 000257                          CCC                    ;CC=1010
013350 000271                          +SEN!SEC
013352 022700 000005                   CMP     #5,R0          ;CC=0101
013356 101002                          BHI     CMP1
013360 102401                          BVS     CMP1
013362 100004                          BPL     CMP2
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 767           <====
013364                          CMP1:
013364 012742 000314                   MOV     #314,-(R2)     ;MOVE TO MAILBOX #  *******  314  *******
013370 005242                          INC     -(R2)          ;SET MSGTYP TO FATAL ERROR
013372 000000                          HALT                   ;CMP DID NOT SET CC'S CORRECTLY
013374 012700 100000            CMP2:  MOV     #100000,R0
013400 000277                          SCC                    ;CC=1101
013402 000242                          CLV
013404 020027 077777                   CMP     R0,#77777      ;CC=0010
013410 101402                          BLOS    CMP3
013412 102001                          BVC     CMP3
013414 100004                          BPL     CMP4
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 752           <====
013416                          CMP3:
013416 012742 000315                   MOV     #315,-(R2)     ;MOVE TO MAILBOX #  *******  315  *******
013422 005242                          INC     -(R2)          ;SET MSGTYP TO FATAL ERROR
013424 000000                          HALT                   ;CMP DID NOT SET CC'S CORRECTLY
013426 052700 040000            CMP4:  BIS     #40000,R0      ;R0=140000
013432 000257                          CCC                    ;CC=0100
013434 000264                          SEZ
013436 022700 040000                   CMP     #40000,R0      ;CC=1011
013442 102003                          BVC     CMP5
013444 103002                          BCC     CMP5
013446 001401                          BEQ     CMP5
013450 100404                          BMI     CMP6
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
```

```
4904                                                    ;           CONDITIONAL BRANCH INST. AND    <====
4905                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
4906                                                    ;           WHICH FOLLOWS W/ 734            <====
4907  013452                            CMP5:
4908  013452  012742  000316                  MOV     #316,-(R2)      ;MOVE TO MAILBOX #  *******  316  *******
4909  013456  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4910  013460  000000                        HALT                    ;CMP DID NOT SET CC'S CORRECTLY
4911  013462  042700  040000        CMP6:   BIC     #40000,R0
4912  013466  000277                        SCC                     ;CC=1111
4913  013470  022700  177777              CMP     #-1,R0          ;CC=0000
4914  013474  101402                        BLOS    CMP7
4915  013476  102401                        BVS     CMP7
4916  013500  100004                        BPL     TST155
4917                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
4918                                                    ;           CONDITIONAL BRANCH INST. AND    <====
4919                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
4920                                                    ;           WHICH FOLLOWS W/ 720            <====
4921  013502                            CMP7:
4922  013502  012742  000317                  MOV     #317,-(R2)      ;MOVE TO MAILBOX #  *******  317  *******
4923  013506  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4924  013510  000000                        HALT                    ;CMP DID NOT SET CC'S CORRECTLY
4925                                                    ;  OR SEQUENCE ERROR
4926
4927                                    ;*****************************************************************************
4928                                    ;TEST 155       TEST COM INSTRUCTION
4929                                    ;*****************************************************************************
4930  013512  005212                    TST155: INC     (R2)            ;UPDATE TEST NUMBER
4931  013514  022712  000155                  CMP     #155,(R2)       ;SEQUENCE ERROR?
4932  013520  001010                        BNE     TST156-10       ;BR TO ERROR HALT ON SEQ ERROR
4933  013522  012700  177777              MOV     #-1,R0
4934  013526  000257                        CCC                     ;CC=1010
4935  013530  000265                        +SEC!SEZ
4936  013532  005100                        COM     R0              ;CC=0101
4937  013534  101002                        BHI     COM1
4938  013536  102401                        BVS     COM1
4939  013540  100004                        BPL     TST156
4940                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
4941                                                    ;           CONDITIONAL BRANCH INST. AND    <====
4942                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
4943                                                    ;           WHICH FOLLOWS W/ 770            <====
4944  013542                            COM1:
4945  013542  012742  000320                  MOV     #320,-(R2)      ;MOVE TO MAILBOX #  *******  320  *******
4946  013546  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
4947  013550  000000                        HALT                    ;COM DID NOT SET CC'S CORRECTLY
4948                                                    ;  OR SEQUENCE ERROR
4949
```

```
                                ;*****************************************************************
                                ;
                                ;       THESE NEXT TWO TESTS VERIFY THE FUNCTIONING OF THE SUB
                                ;AND SBC INSTRUCTIONS.  BOTH OF THESE INSTRUCTIONS HANDLE THE
                                ;C AND V BITS IDENTICALLY.  THE PROCEDURE IS TO PRESET THE CONDITION
                                ;CODES, EXECUTE THE INSTRUCTION WITH A PARTICULAR SET OF DATA, AND
                                ;THEN CHECK THE RESULTS BY EXECUTING A SERIES OF CONDITIONAL
                                ;BRANCHES.  THIS PROCEDURE IS REPEATED SEVERAL TIMES WITH DIFFERENT
                                ;DATA PATTERNS TO PROVIDE EVERY COMBINATION OF THE C AND V BITS.
                                ;
                                ;*****************************************************************
                                ;TEST 156        TEST SUB INSTRUCTION
                                ;*****************************************************************
013552  005212                  TST156: INC     (R2)            ;UPDATE TEST NUMBER
013554  022712  000156                  CMP     #156,(R2)       ;SEQUENCE ERROR?
013560  001055                          BNE     TST157-10       ;BR TO ERROR HALT ON SEQ ERROR
013562  012700  125252                  MOV     #125252,R0
013566  000257                          CCC                     ;CC=1010
013570  000271                          +SEN!SEC
013572  162700  125252                  SUB     #125252,R0      ;CC=0101   R0=0
013576  101002                          BHI     SUB1
013600  102401                          BVS     SUB1
013602  100004                          BPL     SUB2
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                ;           CONDITIONAL BRANCH INST. AND    <====
                                                                ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                ;           WHICH FOLLOWS W/ 767            <====
013604                          SUB1:
013604  012742  000321                  MOV     #321,-(R2)      ;MOVE TO MAILBOX #  *******  321  *******
013610  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
013612  000000                          HALT                    ;SUB DID NOT SET CC'S CORRECTLY
013614  052700  100000          SUB2:   BIS     #100000,R0
013620  000277                          SCC                     ;CC=1101
013622  000242                          CLV
013624  162700  077777                  SUB     #77777,R0       ;CC=0010   R0=1
013630  101402                          BLOS    SUB3
013632  102001                          BVC     SUB3
013634  100004                          BPL     SUB4
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                ;           CONDITIONAL BRANCH INST. AND    <====
                                                                ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                ;           WHICH FOLLOWS W/ 752            <====
013636                          SUB3:
013636  012742  000322                  MOV     #322,-(R2)      ;MOVE TO MAILBOX #  *******  322  *******
013642  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
013644  000000                          HALT                    ;
013646  005100                  SUB4:   COM     R0              ;R0=177777
013650  000277                          SCC                     ;CC=11111

013652  162700  100000                  SUB     #100000,R0      ;CC=0000   R0=77777
013656  101402                          BLOS    SUB5
013660  102401                          BVS     SUB5
013662  100004                          BPL     SUB6
                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                ;           CONDITIONAL BRANCH INST. AND    <====
```

```
                                                        ;               REPLACE THE MOVE INSTRUCTION     <====
                                                        ;               WHICH FOLLOWS W/ 737             <====
        013664                          SUB5:
        013664  012742  000323                  MOV     #323,-(R2)      ;MOVE TO MAILBOX #  *******  323  *******
        013670  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
        013672  000000                          HALT                    ;SUB DID NOT SET CC'S CORRECTLY
        013674  000257                  SUB6:   CCC                     ;CC=0100
        013676  000264                          SEZ
        013700  162700  140000                  SUB     #140000,R0      ;CC=1011
        013704  102003                          BVC     SUB7
        013706  103002                          BCC     SUB7
        013710  001401                          BEQ     SUB7
        013712  100404                          BMI     TST157
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS         <====
                                                        ;           CONDITIONAL BRANCH INST. AND         <====
                                                        ;           REPLACE THE MOVE INSTRUCTION         <====
                                                        ;           WHICH FOLLOWS W/ 723                 <====
        013714                          SUB7:
        013714  012742  000324                  MOV     #324,-(R2)      ;MOVE TO MAILBOX #  *******  324  *******
        013720  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
        013722  000000                          HALT                    ;

                                        ;**************************************************************************
                                        ;TEST 157       TEST SBC INSTRUCTION
                                        ;**************************************************************************
        013724  005212                  TST157: INC     (R2)            ;UPDATE TEST NUMBER
        013726  022712  000157                  CMP     #157,(R2)       ;SEQUENCE ERROR?
        013732  001053                          BNE     TST160-10       ;BR TO ERROR HALT ON SEQ ERROR
        013734  012700  000001                  MOV     #1,R0
        013740  000277                          SCC                     ;CC=1011
        013742  000244                          CLZ
        013744  005600                          SBC     R0              ;CC=0100   R=0
        013746  103403                          BCS     SBC1
        013750  102402                          BVS     SBC1
        013752  100401                          BMI     SBC1
        013754  001404                          BEQ     SBC2
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS         <====
                                                        ;           CONDITIONAL BRANCH INST. AND         <====
                                                        ;           REPLACE THE MOVE INSTRUCTION         <====
                                                        ;           WHICH FOLLOWS W/ 767                 <====
        013756                          SBC1:
        013756  012742  000325                  MOV     #325,-(R2)      ;MOVE TO MAILBOX #  *******  325  *******
        013762  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
        013764  000000                          HALT                    ;SBC DID NOT SET CC'S CORRECTLY
        013766  000277                  SBC2:   SCC                     ;CC=1010
        013770  000245                          +CLZ!CLC
        013772  005600                          SBC     R0              ;CC=0100   R=0
        013774  103403                          BCS     SBC3
        013776  102402                          BVS     SBC3
        014000  100401                          BMI     SBC3
        014002  001404                          BEQ     SBC4
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS         <====
                                                        ;           CONDITIONAL BRANCH INST. AND         <====
                                                        ;           REPLACE THE MOVE INSTRUCTION         <====
                                                        ;           WHICH FOLLOWS W/ 754                 <====
        014004                          SBC3:
```

```
5062  014004  012742  000326          MOV     #326,-(R2)      ;MOVE TO MAILBOX #  *******  326  *******
5063  014010  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5064  014012  000000                  HALT                    ;SBC DID NOT SET CC'S CORRECTLY
5065  014014  000277          SBC4:   SCC                     ;CC=0111
5066  014016  000250                  CLN
5067  014020  005600                  SBC     R0              ;CC=1001   R0=177777
5068  014022  103003                  BCC     SBC5
5069  014024  102402                  BVS     SBC5
5070  014026  001401                  BEQ     SBC5
5071  014030  100404                  BMI     SBC6
5072                                          ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
5073                                          ;           CONDITIONAL BRANCH INST. AND   <====
5074                                          ;           REPLACE THE MOVE INSTRUCTION   <====
5075                                          ;           WHICH FOLLOWS W/ 741           <====
5076  014032          SBC5:
5077  014032  012742  000327          MOV     #327,-(R2)      ;MOVE TO MAILBOX #  *******  327  *******
5078  014036  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5079  014040  000000                  HALT                    ;SBC DID NOT SET CC'S CORRECTLY
5080  014042  042700  077777  SBC6:   BIC     #77777,R0       ;R0=100000
5081  014046  000277                  SCC                     ;CC=1101
5082  014050  000242                  CLV
5083  014052  005600                  SBC     R0              ;CC=0010
5084  014054  101402                  BLOS    SBC7
5085  014056  102001                  BVC     SBC7
5086  014060  100004                  BPL     TST160
5087                                          ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
5088                                          ;           CONDITIONAL BRANCH INST. AND   <====
5089                                          ;           REPLACE THE MOVE INSTRUCTION   <====
5090                                          ;           WHICH FOLLOWS W/ 725           <====
5091  014062          SBC7:
5092  014062  012742  000330          MOV     #330,-(R2)      ;MOVE TO MAILBOX #  *******  330  *******
5093  014066  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5094  014070  000000                  HALT                    ;SBC DID NOT SET CC'S CORRECTLY
5095                                          ;  OR SEQUENCE ERROR
5096
```

```
                                    ;*****************************************************************************
                                    ;
                                    ;         THESE NEXT FOUR TESTS VERIFY THE FUNCTIONING OF THE ROL,
                                    ;ROR, ASL AND ASR INSTRUCTIONS.  SPECIAL DATA PATTERNS ARE LOADED
                                    ;AND ROTATED SEVERAL TIMES FOR EACH TEST.  THE CONDITION CODES
                                    ;ARE PRESET BEFORE EACH ROTATION AND THE CONDITION CODES ARE
                                    ;CHECKED AFTER EACH ROTATION.  THE FINAL CHECK IN EACH TEST IS
                                    ;TO VERIFY THE COMMULATIVE DATA RESULT.  THE DATA PATTERNS HAVE
                                    ;BEEN SELECTED TO PRODUCE ALL COMBINATIONS OF THE C AND V BITS.
                                    ;
                                    ;*****************************************************************************
                                    ;TEST 160        TEST ROL INSTRUCTION
                                    ;*****************************************************************************
014072  005212                      TST160: INC     (R2)            ;UPDATE TEST NUMBER
014074  022712  000160                      CMP     #160,(R2)       ;SEQUENCE ERROR?
014100  001053                              BNE     TST161-10       ;BR TO ERROR HALT ON SEQ ERROR
014102  012700  144000                      MOV     #144000,R0      ;R0=144000
014106  000257                              CCC                     ;CC=0110
014110  000266                              +SEZ!SEV
014112  006100                              ROL     R0              ;CC=1001   R0=110000
014114  103003                              BCC     ROL1
014116  102402                              BVS     ROL1
014120  001401                              BEQ     ROL1
014122  100404                              BMI     ROL2
                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                    ;           CONDITIONAL BRANCH INST. AND    <====
                                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                    ;           WHICH FOLLOWS W/ 767            <====
014124                              ROL1:
014124  012742  000331                      MOV     #331,-(R2)      ;MOVE TO MAILBOX #  *******  331  *******
014130  005242                              INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
014132  000000                              HALT                    ;
014134  000277                      ROL2:   SCC                     ;CC=1100
014136  000243                              +CLV!CLC
014140  006100                              ROL     R0              ;CC=0011   R0=020000
014142  103003                              BCC     ROL3
014144  102002                              BVC     ROL3
014146  001401                              BEQ     ROL3
014150  100004                              BPL     ROL4
                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                    ;           CONDITIONAL BRANCH INST. AND    <====
                                                                    ;           REPLACE THE MOVE INSTRUCTION    <====
                                                                    ;           WHICH FOLLOWS W/ 754            <====
014152                              ROL3:
014152  012742  000332                      MOV     #332,-(R2)      ;MOVE TO MAILBOX #  *******  332  *******
014156  005242                              INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
014160  000000                              HALT                    ;ROL DID NOT SET CC'S CORRECTLY
014162  000277                      ROL4:   SCC                     ;CC=0111
014164  000250                              CLN
014166  006100                              ROL     R0              ;CC=0000   R0=040001
014170  101402                              BLOS    ROL5
014172  102401                              BVS     ROL5
014174  100004                              BPL     ROL6
                                                                    ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                                    ;           CONDITIONAL BRANCH INST. AND    <====
```

```
                                                        ;               REPLACE THE MOVE INSTRUCTION
                                                        ;               WHICH FOLLOWS W/ 742
014176                          ROL5:
014176  012742  000333                  MOV     #333,-(R2)      ;MOVE TO MAILBOX #  *******  333  *******
014202  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
014204  000000                          HALT                    ;ROL DID NOT SET CC'S CORRECTLY
014206  000257                  ROL6:   CCC                     ;CC=0101
014210  000265                          +SEZ!SEC
014212  006100                          ROL     R0              ;CC=1010   R0=100003
014214  101405                          BLOS    ROL7
014216  102004                          BVC     ROL7
014220  100003                          BPL     ROL7
014222  022700  100003                  CMP     #100003,R0
014226  001404                          BEQ     TST161
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                        ;          CONDITIONAL BRANCH INST. AND     <====
                                                        ;          REPLACE THE MOVE INSTRUCTION     <====
                                                        ;          WHICH FOLLOWS W/ 725             <====
014230                          ROL7:
014230  012742  000334                  MOV     #334,-(R2)      ;MOVE TO MAILBOX #  *******  334  *******
014234  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
014236  000000                          HALT                    ;ROL MALFUNCTIONED
                                                        ;  OR SEQUENCE ERROR

                                ;*************************************************************************
                                ;TEST 161       TEST ROR INSTRUCTION
                                ;*************************************************************************
014240  005212                  TST161: INC     (R2)            ;UPDATE TEST NUMBER
014242  022712  000161                  CMP     #161,(R2)       ;SEQUENCE ERROR?
014246  001051                          BNE     TST162-10       ;BR TO ERROR HALT ON SEQ ERROR
014250  012700  000023                  MOV     #23,R0          ;R0=23
014254  000277                          SCC                     ;CC=0111
014256  000250                          CLN
014260  006000                          ROR     R0              ;CC=1001   R0=100011
014262  102403                          BVS     ROR1
014264  103002                          BCC     ROR1
014266  001401                          BEQ     ROR1
014270  100404                          BMI     ROR2
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                        ;          CONDITIONAL BRANCH INST. AND     <====
                                                        ;          REPLACE THE MOVE INSTRUCTION     <====
                                                        ;          WHICH FOLLOWS W/ 767             <====
014272                          ROR1:
014272  012742  000335                  MOV     #335,-(R2)      ;MOVE TO MAILBOX #  *******  335  *******
014276  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
014300  000000                          HALT                    ;ROR DID NOT SET CC'S CORRECTLY
014302  000257                  ROR2:   CCC                     ;CC=1100
014304  000274                          +SEN!SEZ
014306  006000                          ROR     R0              ;CC=0011   R0=040004
014310  102003                          BVC     ROR3
014312  103002                          BCC     ROR3
014314  001401                          BEQ     ROR3
014316  100004                          BPL     ROR4
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS    <====
                                                        ;          CONDITIONAL BRANCH INST. AND     <====
                                                        ;          REPLACE THE MOVE INSTRUCTION     <====
```

```
                                                        ;           WHICH FOLLOWS W/ 754            <===-
014320                          ROR3:
014320  012742  000336                 MOV     #336,-(R2)      ;MOVE TO MAILBOX #  *******  336  *******
014324  005242                         INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
014326  000000                         HALT                    ;ROR DID NOT SET CC'S CORRECTLY
014330  000277                 ROR4:   SCC                     ;CC=1110
014332  000241                         CLC
014334  006000                         ROR     R0              ;CC=0000   R0=020002
014336  101403                         BLOS    ROR5
014340  102402                         BVS     ROR5
014342  001401                         BEQ     ROR5
014344  100004                         BPL     ROR6
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 741           <====
014346                          ROR5:
014346  012742  000337                 MOV     #337,-(R2)      ;MOVE TO MAILBOX #  *******  337  *******
014352  005242                         INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
014354  000000                         HALT                    ;ROR DID NOT SET CC'S CORRECTLY
014356  000257                 ROR6:   CCC                     ;CC=0101
014360  000265                         +SEC!SEZ
014362  006000                         ROR     R0              ;CC=1010   R0=110001
014364  101402                         BLOS    ROR7
014366  102001                         BVC     ROR7
014370  100404                         BMI     TST162
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 727           <====
014372                          ROR7:
014372  012742  000340                 MOV     #340,-(R2)      ;MOVE TO MAILBOX #  *******  340  *******
014376  005242                         INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
014400  000000                         HALT                    ;ROR DID NOT PRODUCE CORRECT RESULTS
                                                        ;  OR SEQUENCE ERROR

                                ;*****************************************************************************
                                ;TEST 162       TEST ASL INSTRUCTION
                                ;*****************************************************************************
014402  005212                 TST162: INC     (R2)            ;UPDATE TEST NUMBER
014404  022712  000162                 CMP     #162,(R2)       ;SEQUENCE ERROR?
014410  001054                         BNE     TST163-10       ;BR TO ERROR HALT ON SEQ ERROR
014412  012700  144000                 MOV     #144000,R0      ;R0=14000
014416  000257                         CCC                     ;CC=0110
014420  000271                         +SEN!SEC
014422  006300                         ASL     R0              ;CC=1001   R0=110000
014424  103003                         BCC     ASL1
014426  102402                         BVS     ASL1
014430  001401                         BEQ     ASL1
014432  100404                         BMI     ASL2
                                                        ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
                                                        ;           CONDITIONAL BRANCH INST. AND   <====
                                                        ;           REPLACE THE MOVE INSTRUCTION   <====
                                                        ;           WHICH FOLLOWS W/ 767           <====
014434                          ASL1:
014434  012742  000341                 MOV     #341,-(R2)      ;MOVE TO MAILBOX #  *******  341  *******
```

```
5265  014440  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5266  014442  000000                        HALT                    ;
5267  014444  000277                ASL2:   SCC                     ;CC=1100
5268  014446  000243                        +CLV!CLC
5269  014450  006300                        ASL     R0              ;CC=0011   R0=020000
5270  014452  103003                        BCC     ASL3
5271  014454  102002                        BVC     ASL3
5272  014456  001401                        BEQ     ASL3
5273  014460  100004                        BPL     ASL4
5274                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
5275                                                                ;           CONDITIONAL BRANCH INST. AND   <====
5276                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
5277                                                                ;           WHICH FOLLOWS W/ 754           <====
5278  014462                        ASL3:
5279  014462  012742  000342                MOV     #342,-(R2)      ;MOVE TO MAILBOX #  *******  342  *******
5280  014466  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5281  014470  000000                        HALT                    ;ASL DID NOT SET CC'S CORRECTLY
5282  014472  000277                ASL4:   SCC                     ;CC=0111
5283  014474  000250                        CLN
5284  014476  006300                        ASL     R0              ;CC=0000   R0=040000
5285  014500  101402                        BLOS    ASL5
5286  014502  102401                        BVS     ASL5
5287  014504  100004                        BPL     ASL6
5288                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
5289                                                                ;           CONDITIONAL BRANCH INST. AND   <====
5290                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
5291                                                                ;           WHICH FOLLOWS W/ 742           <====
5292  014506                        ASL5:
5293  014506  012742  000343                MOV     #343,-(R2)      ;MOVE TO MAILBOX #  *******  343  *******
5294  014512  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5295  014514  000000                        HALT                    ;ASL DID NOT SET CC'S CORRECTLY
5296  014516  000257                ASL6:   CCC                     ;CC=0101
5297  014520  000265                        +SEZ!SEC
5298  014522  006300                        ASL     R0              ;CC=1010   R0=100000
5299  014524  103406                        BCS     ASL7
5300  014526  001405                        BEQ     ASL7
5301  014530  102004                        BVC     ASL7
5302  014532  100003                        BPL     ASL7
5303  014534  022700  100000                CMP     #100000,R0
5304  014540  001404                        BEQ     TST163
5305                                                                ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
5306                                                                ;           CONDITIONAL BRANCH INST. AND   <====
5307                                                                ;           REPLACE THE MOVE INSTRUCTION   <====
5308                                                                ;           WHICH FOLLOWS W/ 724           <====
5309  014542                        ASL7:
5310  014542  012742  000344                MOV     #344,-(R2)      ;MOVE TO MAILBOX #  *******  344  *******
5311  014546  005242                        INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5312  014550  000000                        HALT                    ;ASL MALFUNCTIONED
5313                                                                ;  OR SEQUENCE ERROR
5314
5315                                ;*****************************************************************************
5316                                ;TEST 163       TEST ASR INSTRUCTION
5317                                ;*****************************************************************************
5318  014552  005212                TST163: INC     (R2)            ;UPDATE TEST NUMBER
5319  014554  022712  000163                CMP     #163,(R2)       ;SEQUENCE ERROR?
5320  014560  001060                        BNE     TST164-10       ;BR TO ERROR HALT ON SEQ ERROR
```

```
 5321  014562  012700  100023          MOV     #100023,R0      ;R0=100023
 5322  014566  000277                  SCC                     ;CC=0110
 5323  014570  000250                  CLN
 5324  014572  006200                  ASR     R0              ;CC=1001   RP=140011
 5325  014574  102403                  BVS     ASR1
 5326  014576  103002                  BCC     ASR1
 5327  014600  001401                  BEQ     ASR1
 5328  014602  100404                  BMI     ASR2
 5329                                          ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 5330                                          ;           CONDITIONAL BRANCH INST. AND   <====
 5331                                          ;           REPLACE THE MOVE INSTRUCTION   <====
 5332                                          ;           WHICH FOLLOWS W/ 767           <====
 5333  014604                  ASR1:
 5334  014604  012742  000345          MOV     #345,-(R2)      ;MOVE TO MAILBOX #  *******  345  *******
 5335  014610  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 5336  014612  000000                  HALT                    ;ASR DID NOT SET CC'S CORRECTLY
 5337  014614  042700  100000  ASR2:   BIC     #100000,R0      ;R0=40011
 5338  014620  000277                  SCC                     ;CC=1100
 5339  014622  000243                  +CLV!CLC
 5340  014624  006200                  ASR     R0              ;CC=0011   R0=020004
 5341  014626  102003                  BVC     ASR3
 5342  014630  103002                  BCC     ASR3
 5343  014632  001401                  BEQ     ASR3
 5344  014634  100004                  BPL     ASR4
 5345                                          ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 5346                                          ;           CONDITIONAL BRANCH INST. AND   <====
 5347                                          ;           REPLACE THE MOVE INSTRUCTION   <====
 5348                                          ;           WHICH FOLLOWS W/ 752           <====
 5349  014636                  ASR3:
 5350  014636  012742  000346          MOV     #346,-(R2)      ;MOVE TO MAILBOX #  *******  346  *******
 5351  014642  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 5352  014644  000000                  HALT                    ;ASR DID NOT SET CC'S CORRECTLY
 5353  014646  000277          ASR4:   SCC                     ;CC=1111
 5354
 5355  014650  006200                  ASR     R0              ;CC=0000   R0=010002
 5356  014652  101403                  BLOS    ASR5
 5357  014654  102402                  BVS     ASR5
 5358  014656  001401                  BEQ     ASR5
 5359  014660  100004                  BPL     ASR6
 5360                                          ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
 5361                                          ;           CONDITIONAL BRANCH INST. AND   <====
 5362                                          ;           REPLACE THE MOVE INSTRUCTION   <====
 5363                                          ;           WHICH FOLLOWS W/ 740           <====
 5364  014662                  ASR5:
 5365  014662  012742  000347          MOV     #347,-(R2)      ;MOVE TO MAILBOX #  *******  347  *******
 5366  014666  005242                  INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
 5367  014670  000000                  HALT                    ;ASR DID NOT SET CC'S CORRECTLY
 5368  014672  052700  100000  ASR6:   BIS     #100000,R0      ;R0=110002
 5369  014676  000257                  CCC                     ;CC=0101
 5370  014700  000265                  +SEZ!SEC
 5371  014702  006200                  ASR     R0              ;C=1010   R0=144001
 5372  014704  101406                  BLOS    ASR7
 5373  014706  102005                  BVC     ASR7
 5374  014710  100004                  BPL     ASR7
 5375  014712  001403                  BEQ     ASR7
 5376  014714  022700  144001          CMP     #144001,R0      ;CHECK RESULT OF ASR'S
```

```
MAINDEC-11-DGKAA  11 04 CPU TEST        MACY11 27(732)  06-OCT-76  14:01  PAGE 275
DGKAAA.P11       *163    TEST ASR INSTRUCTION

  5377  014720  001404                          BEQ     TST164
  5378                                                          ; TO SCOPE: CLEAR THE RIGHT BYTE OF THIS   <====
  5379                                                          ;           CONDITIONAL BRANCH INST. AND   <====
  5380                                                          ;           REPLACE THE MOVE INSTRUCTION   <====
  5381                                                          ;           WHICH FOLLOWS W/ 720           <====
  5382  014722                          ASR7:
  5383  014722  012742  000350                    MOV     #350,-(R2)      ;MOVE TO MAILBOX #  *******  350  *******
  5384  014726  005242                          INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
  5385  014730  000000                          HALT                    ;ASR DID NOT FUNCTION CORRECTLY
  5386                                                                  ;  OR SEQUENCE ERROR
  5387
  5388
```

```
                                ;*****************************************************************************
                                ;
                                ;       THIS TEST VERIFIES THE CONTENTS OF THE BRANCH ROM.  THE TEST
                                ;EXECUTES EVERY POSSIBLE BRANCH WITH EVERY POSSIBLE CONDITION
                                ;CODE COMBINATION.
                                ;       THE ROUTINE USES TWO TABLES.  THE BRANCH TABLE HOLDS ALL THE
                                ;POSSIBLE BRANCH INSTRUCTIONS, THE OTHER TABLE (YNTAB) HOLDS BIT MAPS FOR
                                ;EACH BRANCH.  A ONE IN THE BIT MAP INDICATES THAT THE CORRESPONDING
                                ;BRANCH INSTRUCTION SHOULD BRANCH FOR THE CONDITION CODE SETTING WHICH
                                ;CORRESPONDS TO THE BIT POSITION WITHIN THE MAP.  FOR EXAMPLE IF THE LEFT
                                ;MOST BIT IS A ONE THEN THE CORRESPONDING BRANCH INSTRUCTION SHOULD BRANCH
                                ;WHEN THE CONDITION CODES ARE 0.
                                ;       THE ROUTINE CONSISTS OF NESTED LOOPS; THE OUTTER LOOP SETS UP
                                ;ALL THE POSSIBLE BRANCH INSTRUCTIONS.  THE INNER LOOP SETS UP EVERY POSSIBLE
                                ;CONDITION CODE FOR EACH BRANCH.
                                ;       THE BIT MAP IS USED TO SET THE ADDRESS LOCATION IN TWO
                                ;JUMP MODE 3 INSTRUCTIONS.  THE ADDRESSES ARE CHANGED TO ALLOW THE
                                ;PROGRAM TO CONTINUE OR JUMP TO AN ERROR ROUTINE DEPENDING UPON
                                ;WHETHER IT HANDLED THE BRANCH INSTRUCTION CORRECTLY.
                                ;       AT ANY ERROR HALT, LOCATION, BRH, HOLDS THE BRANCH INSTRUCTION
                                ;UNDER TEST AND LOCATION, CC, HOLDS THE VALUE OF THE CONDITION CODES
                                ;AT THE TIME THE BRANCH WAS EXECUTED.
                                ;
                                ;*****************************************************************************
                                ;TEST 164       TEST THE BRANCH ROM
                                ;*****************************************************************************
014732  005212                  TST164: INC     (R2)            ;UPDATE TEST NUMBER
014734  022712  000164                  CMP     #164,(R2)       ;SEQUENCE ERROR?
014740  001062                          BNE     ER              ;BR TO ERROR HALT ON SEQ ERROR
014742  012700  015332          SETUP:  MOV     #BRTAB,R0       ;INITIALIZE BRANCH TABLE POINTER
014746  012704  015370                  MOV     #YNTAB,R4       ;INITIALIZE YES/NO BRANCH MAP POINTER
014752  012767  000015  000142          MOV     #15,BRCT        ;INITIALIZE BRANCH TABLE COUNT
014760  012067  000110          SETBR:  MOV     (R0)+,BRH       ;GET NEXT BRANCH INST.
014764  012401                          MOV     (R4)+,R1        ;GET NEXT BRANCH MAP
014766  012767  177777  000074          MOV     #-1,CC          ;INITIALIZE CONDITION CODE VALUE
014774  012703  000016                  MOV     #16,R3          ;INITIALIZE CONDITION CODE COUNT
015000  005267  000064          SETCC:  INC     CC              ;SET FOR NEXT CC VALUE
015004  032701  100000                  BIT     #100000,R1      ;SEE IF SHOULD BR W/ THESE CC'S
015010  013705  177776                  MOV     @#177776,R5     ;SIMULATE A JNE
015014  042705  177773                  BIC     #177773,R5      ;       (JUMP NOT EQUAL)
015020  000165  015024                  JMP     .+4(R5)         ;       TO SET2BR
015024  000167  000020                  JMP     SET2BR
015030  012767  015124  000042          MOV     #CONT,NBR       ;SET TO CONTINUE IF NO BRANCH
015036  012767  015106  000040          MOV     #ER,YBR         ;SET TO REPORT ERROR IF BRANCH
015044  000167  000014                  JMP     AROUND          ;GO AROUDND OPPOSITE CONDITION
015050  012767  015106  000022  SET2BR: MOV     #ER,NBR         ;SET TO REPORT ERROR IF NO BRANCH
015056  012767  015124  000020          MOV     #CONT,YBR       ;SET TO CONTINUE IF BRANCH
015064  006101                  AROUND: ROL     R1              ;UPDATE BIT MAP
                                ;
015066  012737                          MOV     (PC)+,@(PC)+    ;SET CONDITION CODE
015070  000000          CC:     0                               ;NEW CC VALUE GOES HERE
015072  177776                  177776
015074  000000          BRH:    0                               ;BRANCH INST. GOES HERE
015076  000137                  JMP     @(PC)+                  ;THIS JUMP IF NO BRANCH
015100  000000          NBR:    0                               ;WHERE TO GO IF NO BRANCH OCCURS
```

```
5445  015102  000137                    JMP     @(PC)+          ;THIS JUMP IF BRANCH OCCURS
5446  015104  000000            YBR:    0                       ;WHERE TO GO IF BRANCH OCCURS
5447  015106  012702  000304    ER:     MOV     #$TESTN,R2      ;RESTORE POINTER
5448  015112  012742  000351            MOV     #351,-(R2)      ;MOVE TO MAILBOX #  *******  351  *******
5449  015116  005242                    INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5450  015120  000000                    HALT                    ;
5451  015122  000000            BRCT:   0
5452  015124  005303            CCNT:   DEC     R3              ;CC'S DONE?
5453  015126  013705  177776            MOV     @#177776,R5     ;SIMULATE A JNE
5454  015132  042705  177773            BIC     #177773,R5      ;       (JUMP NOT EQUAL)
5455  015136  000165  015142            JMP     .+4(R5)         ;       TO SETCC
5456  015142  000167  177632            JMP     SETCC
5457  015146  005367  177750            DEC     BRCT            ;BR'S DONE?
5458  015152  013705  177776            MOV     @#177776,R5     ;SIMULATE A JNE
5459  015156  042705  177773            BIC     #177773,R5      ;       (JUMP NOT EQUAL)
5460  015162  000165  015166            JMP     .+4(R5)         ;       TO SETBR
5461  015166  000167  177566            JMP     SETBR
```

```
5462                                    ;*****************************************************************************
5463                                    ;TEST 165       END OF PASS SEQUENCE
5464                                    ;*****************************************************************************
5465  015172  005212                    TST165: INC     (R2)            ;UPDATE TEST NUMBER
5466  015174  022712  000165                    CMP     #165,(R2)       ;SEQUENCE ERROR?
5467  015200  001037                            BNE     EOP1            ;BR TO ERROR HALT ON SEQ ERROR
5468  015202  005237  000306                    INC     @#$PASS
5469  015206  105267  000076                    INCB    PASSPT          ;SHOULD PRINT THIS PASS?
5470  015212  001020                            BNE     ACT             ;NO
5471  015214  132767  000040  163077            BITB    #40,$ENVM       ;WILL APT ALLOW PRINTING?
5472  015222  001014                            BNE     ACT             ;NO
5473  015224  023727  000042  015264            CMP     @#42,#SENDAD    ;UNDER ACT AUTO ACCEPT?
5474  015232  001410                            BEQ     ACT             ;IF SO SKIP PRINTOUT
5475  015234  012700  015312                    MOV     #MSG,R0         ;GET MSG ADDR.
5476  015240  105737  177564            WAIT:  TSTB    @#TPS           ;TTY READY
5477  015244  100375                            BPL     WAIT            ;NO WAIT
5478  015246  112037  177566                    MOVB    (R0)+,@#TPB     ;PRINT CHARACTER
5479  015252  001372                            BNE     WAIT            ;NEXT IF NOT DONE.
5480  015254  013700  000042            ACT:   MOV     @#42,R0         ;CHECK ACT
5481  015260  001405                            BEQ     GOAGIN          ;KEEP GOING
5482  015262  000005                            RESET
5483  015264  004710                    SENDAD: JSR     PC,(R0)         ;ACT HOOKS
5484  015266  000240                            NOP
5485  015270  000240                            NOP
5486  015272  000240                            NOP
5487  015274  000167  163222            GOAGIN: JMP     RESTRT          ;DO NEXT PASS
5488  015300                            EOP1:
5489  015300  012742  000352                    MOV     #352,-(R2)      ;MOVE TO MAILBOX #  *******  352  *******
5490  015304  005242                            INC     -(R2)           ;SET MSGTYP TO FATAL ERROR
5491  015306  000000                            HALT                    ;SEQUENCE ERROR
5492  015310  177777                    PASSPT: -1
5493  015312  047105  020104  043117    MSG:   .ASCIZ  .END OF DGKAA.<15><12>
5494  015320  042040  045507  040501
5495  015326  005015      000
```

```
5496                015332                  .EVEN
5497
5498  015332  000402                BRTAB:  BR      .+6
5499  015334  001002                        BNE     .+6
5500  015336  001402                        BEQ     .+6
5501  015340  002002                        BGE     .+6
5502  015342  002402                        BLT     .+6
5503  015344  003002                        BGT     .+6
5504  015346  003402                        BLE     .+6
5505  015350  100002                        BPL     .+6
5506  015352  100402                        BMI     .+6
5507  015354  101002                        BHI     .+6
5508  015356  101402                        BLOS    .+6
5509  015360  102002                        BVC     .+6
5510  015362  102402                        BVS     .+6
5511  015364  103002                        BCC     .+6
5512  015366  103402                        BCS     .+6
5513
5514              000002                    .RADIX  2
5515  015370  177777                YNTAB:  1111111111111111        ;BR
5516  015372  170360                        1111000011110000        ;BNE:  Z=0
5517  015374  007417                        0000111100001111        ;BEQ:  Z=1
5518  015376  146063                        1100110000110011        ;BGE:  N XOR V =0
5519  015400  031714                        0011001111001100        ;BLT:  N XOR V =1
5520  015402  140060                        1100000000110000        ;BGT:  Z+(N XOR V) =0
5521  015404  037717                        0011111111001111        ;BLE:  Z+(N XOR V) =1
5522
5523  015406  177400                        1111111100000000        ;BPL:  N=0
5524  015410  000377                        0000000011111111        ;BMI:  N=1
5525  015412  120240                        1010000010100000        ;BHI:  C+Z=0
5526  015414  057537                        0101111101011111        ;BLOS: C+Z=1
5527  015416  146314                        1100110011001100        ;BVC:  V=0
5528  015420  031463                        0011001100110011        ;BVS:  V=1
5529  015422  125252                        1010101010101010        ;BCC:  C=0
5530  015424  052525                        0101010101010101        ;BCS:  C=1
5531
5532              000010                    .RADIX  8
5533  015426  012737  015436  000024  PWRDN:  MOV     #PWRUP,@#24             ;SET UP FOR A POWER UP
5534  015434  000000                        HALT
5535
5536  015436  012737  015426  000024  PWRUP:  MOV     #PWRDN,@#24             ;SET UP FOR A POWER FAIL
5537  015444  012706  000500                MOV     #STBOT,R6               ;SET UP STACK POINTER
5538  015450  132767  000040  162643        BITB    #40,$ENVM               ;SHOULD PRINT?
5539  015456  001010                        BNE     PWR2                    ;IF NOT: BR
5540  015460  012700  015504                MOV     #PFMES,R0               ;GET POWER FAIL MESSG.
5541  015464  105737  177564        WATE:   TSTB    @#TPS                   ;TTY READY?
5542  015470  100375                        BPL     WATE                    ;IF NOT: BR
5543  015472  112037  177566                MOVB    (R0)+,@#TPB             ;PRINT NEXT CHAR.
5544  015476  001372                        BNE     WATE                    ;IF NOT DONE:  BR
5545  015500  000137  000500        PWR2:   JMP     @#START                 ;START PROGRAM AGAIN
5546
5547  015504  006412  047520  042527  PFMES:  .ASCIZ  <12><15>.POWER FAILURE.<12><15>
5548  015512  020122  040506  046111
5549  015520  051125  005105  000015
5550
5551              00000:                    .END
```

```
ABASE = 0C0000          424
ACDW1 = 000000          424
ACDW2 = 000000          424
ACPUOP= 000000          424     439
ACT     015254         5470    5472    5474    5480#
ADC1    013120         4750    4751    4757#
ADC2    013130         4752    4761#
ADC3    013150         4765    4766    4772#
ADC4    013160         4767    4776#
ADC5    013176         4779    4780    4781    4787#
ADDM0 = 000000          424
ADDM1 = 000000          424
ADDM10= 000000          424
ADDM11= 000000          424
ADDM12= 000000          424
ADDM13= 000000          424
ADDM14= 000000          424
ADDM15= 000000          424
ADDM2 = 000000          424
ADDM3 = 000000          424
ADDM4 = 000000          424
ADDM5 = 000000          424
ADDM6 = 000000          424
ADDM7 = 000000          424
ADDM8 = 000000          424
ADDM9 = 000000          424
ADD1    012734         4672    4673    4679#
ADD2    012744         4674    4683#
ADD3    012760         4686    4687    4693#
ADD4    012770         4688    4697#
ADD5    013006         4700    4701    4707#
ADD6    013016         4702    4711#
ADD7    013030         4712    4713    4719#
ADD8    013040         4714    4723#
ADD9    013060         4726    4727    4728    4734#
ADEVCT= 000000          424     430
ADEVM = 000000          424
AENV  = 000000          424     435
AENVM = 000000          424     436
AFATAL= 000000          424     427
AMADR1= 000000          424
AMADR2= 000000          424
AMADR3= 000000          424
AMADR4= 000000          424
AMAMS1= 000000          424
AMAMS2= 000000          424
AMAMS3= 000000          424
AMAMS4= 000000          424
AMSGAD= 000000          424     432
AMSGLG= 000000          424     433
AMSGTY= 000000          424     426
AMTYP1= 000000          424
AMTYP2= 000000          424
AMTYP3= 000000          424
AMTYP4= 000000          424
APASS = 000000          424     429
```

```
APRIOR= 000000            424
AROUND  015064           5434    5437*
ASL1    014434           5255    5256    5257    5263*
ASL2    014444           5258    5267*
ASL3    014462           5270    5271    5272    5278*
ASL4    014472           5273    5282*
ASL5    014506           5285    5286    5292*
ASL6    014516           5287    5296*
ASL7    014542           5299    5300    5301    5302    5309*
ASR1    014604           5325    5326    5327    5333*
ASR2    014614           5328    5337*
ASR3    014636           5341    5342    5343    5349*
ASR4    014646           5344    5353*
ASR5    014662           5356    5357    5358    5364*
ASR6    014672           5359    5368*
ASR7    014722           5372    5373    5374    5375    5382*
ASWREG= 000000            424     437
ATESTN= 000000            424     429
AUNIT = 000000            424     431
AUSWR = 000000            424     438
AVECT1= 000000            424
AVECT2= 000000            424
BIC1    012064           4345    4346    4352*
BIC2    012074           4347    4356*
BIC3    012112           4359    4360    4366*
BIS1    012154           4382    4383    4384    4390*
BIS2    012164           4385    4394*
BIS3    012204           4397    4398    4399    4405*
BIT1    011774           4307    4308    4314*
BIT2    012004           4309    4319*
BIT3    012022           4322    4323    4329*
BRCT    015122           5421*   5451*   5457*
BRC1    002350           1228    1234*
BRC2    002360           1229    1239*
BRC3    002370           1241    1247*
BRH     015074           5422*   5442*
BRN1    002230           1134    1140*
BRN2    002240           1135    1145*
BRN3    002250           1147    1153*
BRTAB   015332           5419    5498*
BRV1    002300           1181    1187*
BRV2    002310           1182    1192*
BRV3    002320           1194    1200*
BRZ1    002160           1087    1093*
BRZ2    002170           1088    1098*
BRZ3    002200           1100    1106*
BR1     000572            504     510*
BR2     000602            505     514*
BR3     000614            515     523*
BR4     000622            524     530*
BR5     000632            525     534*
CC      015070           5424*   5426*   5440*
CLR1    012522           4563    4564    4565    4571*
CMP1    013364           4869    4870    4876*
CMP2    013374           4871    4880*
CMP3    013416           4884    4885    4891*
```

```
CMP4    013426          4886    4895#
CMP5    013452          4899    4900    4901    4907#
CMP6    013462          4902    4911#
CMP7    013502          4914    4915    4921#
COM1    013542          4937    4938    4944#
CONT    015124          5432    5436    5452#
DEC1    012362          4486    4487    4488    4494#
DEC2    012372          4489    4498#
DEC3    012406          4501    4502    4508#
DEC4    012416          4503    4512#
DEC5    012432          4515    4516    4522#
DEC6    012442          4517    4526#
DEC7    012464          4530    4531    4532    4538#
DNM1    005536          2707    2715#
DNM2    005552          2716    2724#
DNM3    005570          2726    2734#
DNM4    005612          2738    2746#
DOP82A  006212          2977    2985#
DOP82B  006270          3015    3023#
DOPOA   005370          2631    2639#
DOPOB   005414          2644    2652#
DOPOC   005434          2655    2663#
DOPOD   005464          2670    2678#
DOP1    006024          2869    2877#
DOP2    006136          2937    2945#
DOP4    006516          3134    3146#
DOP5    006602          3172    3184#
EOP1    015300          5467    5488#
ER      015106          5418    5433    5435    5447#
GOAGIN  015274          5481    5487#
IJMP    011100          4031    4034#
IJMP4   010652          3956    3959#   3973
IJMP5   011042          4017    4020#
INC1    012244          4433    4434    4440#
INC2    012254          4435    4444#
INC3    012276          4448    4449    4450    4456#
INC4    012306          4451    4461#
INC5    012322          4464    4465    4471#
JMPCK   011102          3934    4034    4036#
JMPSEQ  011122          3935#   3947    3957#   3961    3970#   3982    3992#   3995    4005#   4008    4018#   4022    4032#
                        4036    4046#
JMP2    010654          3936    3961#
JMP2A   010672          3962    3970#
JMP3    010602          3938#   3971
JMP3A   010620          3939    3947#
JMP3B   010640          3948    3956#
JMP4    010704          3959    3973#
JMP4A   010722          3974    3982#
JMP4B   010742          3983    3991#
JMP5    011010          3991    4008#
JMP5A   011030          4009    4017#
JMP6    010754          3995#   4020
JMP6A   010774          3996    4004#
JMP7    011044          4004    4022#
JMP7A   011064          4023    4031#
JSRCK   011554          4195    4198#
```

```
JSRCKA  011550          4192    4195#
JSRCK1  011572          4071    4199    4206#
JSRSEQ  011552          4075#   4086    4104#   4107    4125#   4129    4141#   4145    4159#   4161    4175#   4179    4195#
                        4196#   4198
JSR0    011136          4069    4071#
JSR1    011142          4070    4073#
JSR1A   011164          4081#   4109
JSR2    011254          4074    4107#   4131
JSR2A   011304          4108    4110    4112    4119#
JSR2B   011314          4114    4123#
JSR3    011174          4086#   4094    4126
JSR3A   011234          4087    4089    4091    4093    4100#
JSR3B   011244          4095    4104#
JSR4    011336          4088    4105    4129#
JSR4A   011354          4130    4137#
JSR4B   011364          4132    4141#
JSR5    011442          4142    4161#   4165    4181
JSR5A   011466          4162    4164    4171#
JSR5B   011476          4166    4175#
JSR6    011376          4145#   4163    4194
JSR6A   011422          4146    4148    4155#
JSR6AD  011546          4149    4176    4194#   4200
JSR6B   011432          4150    4159#
JSR7    011510          4147    4160    4179#
JSR7A   011526          4180    4187#
JSR7B   011536          4182    4191#
MOV1    011704          4269    4270    4276#
MOV2    011714          4271    4281#
MOV3    011732          4284    4285    4291#
MSG     015312          5475    5493#
NBR     015100          5432#   5435#   5444#
NEG1    013240          4813    4814    4815    4821#
NEG2    013250          4816    4825#
NEG3    013272          4829    4830    4831    4837#
NEG4    013302          4832    4841#
NEG5    013322          4845    4846    4847    4853#
PASSPT  015310           491#   5469#   5492#
PC     =%000007          483    5439#   5443    5445    5483#
PFMES   015504          5540    5547#
PS     = 177776          406#    874#    875     892#    893     910#    911     928#    929
PWRDN   015426           489    5533#   5536
PWRUP   015436          5533    5536#
PWR2    015500          5539    5545#
REG1    001216           739#    740
REG2    001252           759#    760
REG2A   001274           756     761     770#
REG3    001316           779#    780
REG4    001352           799#    800
REG5    001406           819#    820
REG6    001442           839#    840
RESTRT  000522           492#   5487
ROL1    014124          5118    5119    5120    5126#
ROL2    014134          5121    5130#
ROL3    014152          5133    5134    5135    5141#
ROL4    014162          5136    5145#
ROL5    014176          5148    5149    5155#
```

```
ROL6    014206          5150    5159*
ROL7    014230          5162    5163    5164    5171*
ROR1    014272          5187    5188    5189    5195*
ROR2    014302          5190    5199*
ROR3    014220          5202    5203    5204    5210*
ROR4    014330          5205    5214*
ROR5    014346          5217    5218    5219    5225*
ROR6    014356          5220    5229*
ROR7    014372          5232    5233    5239*
ROTX    007710          3545    3547*   3551    3564*   3579*   3581*   3585    3608*   3609    3613
ROTXAD  010034          3609*   3611*   3624*
ROTOA   007026          3276    3277    3284*
ROTOB   007036          3279    3288*
ROTOC   007060          3291    3292    3299*
ROT1A   007126          3324    3325    3332*
ROT1B   007136          3327    3336*
ROT1C   007162          3339    3340    3347*
ROT1D   007172          3342    3351*
ROT1E   007222          3356    3357    3364*
ROT2A   007274          3387    3389    3397*
ROT2B   007304          3392    3401*
ROT2C   007334          3405    3407    3414*
ROT2D   007344          3409    3418*
ROT2E   007400          3423    3425    3433*
ROT3A   007446          3455    3462*
ROT3B   007456          3457    3466*
ROT3C   007504          3469    3476*
ROT3D   007514          3471    3480*
ROT3E   007542          3483    3490*
ROT4    007616          3514    3516    3523*
ROT5    007700          3544    3550    3552-   3559*
ROT6    007750          3582    3589*
ROT7    010024          3607    3612    3619*
RTS1    011640          4228    4235*
R0      =%000000         665*    666     683*    684     701*    702     719*    720     1294*   1303*   1304*   1313*   1340*
                        1341*   1351*   1354*   1356*   1357*   1381*   1390*   1392*   1418*   1419*   1428*   1431*   1456*
                        1457*   1458*   1459*   1468*   1470*   1472*   1500*   1501*   1502*   1503*   1504*   1513*   1514*
                        1515*   1516*   1518*   1546*   1547*   1548*   1549*   1550*   1551*   1560*   1561*   1562*   1564*
                        1565*   1566*   1594*   1595*   1596*   1597*   1598*   1599*   1608*   1609*   1610*   1612*   1613*
                        1636*   1637*   1638*   1639*   1640*   1641*   1642*   1651*   1652*   1653*   1654*   1655*   1657*
                        1658*   1690*   1691*   1692*   1693*   1694*   1703*   1704*   1705*   1707*   1740*   1741*   1742*
                        1791*   1792*   1793*   1794*   1795*   1796*   1805*   1806*   1807*   1808*   1809*   1810*   1812*
                        1830*   1831*   1832*   1833*   1842*   1843*   1844*   1846*   1847*   1848*   1881*   1882*   1883*
                        1884*   1893*   1894*   1895*   1897*   1923*   1924*   1925*   1926*   1935*   1937*   1964*   1965*
                        1966*   1967*   1976*   1978*   2044*   2047    2076*   2077*   2080    2108*   2109*   2112    2140*
                        2141*   2142*   2145    2157*   2158*   2161    2189*   2190*   2193    2206*   2207*   2232*   2233*
                        2234*   2237    2249*   2258*   2261    2274*   2275*   2299*   2300*   2301*   2302*   2305    2318*
                        2319*   2345*   2346*   2347*   2348*   2349*   2350    2353    2367    2380    2381    2405*   2406*
                        2407*   2410    2422    2446*   2447*   2448*   2449*   2450*   2453    2465*   2466*   2490*   2491*
                        2492*   2493*   2496    2508*   2532*   2533*   2534*   2535*   2538    2550*   2573*   2574*   2576
                        2600*   2602*   2603    2629*   2630    2639*   2640*   2642    2652    2663*   2664*   2667    2668
                        2678    2703*   2706    2715    2724*   2725    2734*   2735*   2737    2747    2771*   2772*   2776*
                        2799*   2800*   2801*   2803    2828*   2829*   2830*   2833    2861*   2862*   2863*   2864*   2867
                        2877*   2878    2902*   2903*   2907*   2908*   2932*   2933*   2934*   2935    2945*   2946*   2972*
                        2973*   2974*   2975*   3008*   3010*   3011*   3012*   3013    3023    3024    3050*   3051*   3052*
                        3077*   3078*   3079    3104*   3105*   3106    3135*   3136    3137    3138    3139    3140    3173*
```

CROSS REFERENCE TABLE -- USER SYMBOLS

```
                3174    3175    3176    3177    3178    3211*   3212    3213    3214    3215    3216    3243*   [illegible]
                3245    3246    3247    3248    3273*   3275*   3278    3288*   3290*   3293    3320*   332.*   3323*
                3337*   3338*   3351*   3352*   3353*   3355*   3383*   3384*   3386*   3390*   3391*   3401*   3402*
                3404*   3408*   3418*   3419*   3420*   3422*   3426*   3427*   3511*   3513*   3517    3546*   3549*
                3553    3640*   3641*   3650    3675*   3676*   3703*   3704*   3714*   3767*   3768*   3778    3805*
                3807*   3817    3848*   3849*   3850    3881*   3882*   3883    3936*   3937    3938    3956*   3958
                3971*   3972    3973    3991*   3993    4004*   4006    4017*   4019    4031*   4033    4074*   4078
                4094    4126*   4127    4142*   4143    4149    4165    4176*   4177    4228*   4229*   4235    4268*
                4283*   4303*   4306    4321    4341*   4344*   4358*   4378*   4381*   4396*   4429*   4432*   4444*
                4447*   4463*   4483*   4485*   4500*   4514*   4526*   4529*   4562*   4585    4598*   4601    4621*
                4624*   4638*   4669*   4671*   4685*   4699*   4711*   4725*   4746*   4749*   4761*   4764*   4778*
                4809*   4812*   4825*   4828*   4841*   4844*   4865*   4868    4880*   4883    4895*   4898    4911*
                4913    4933*   4936*   4967*   4970*   4982*   4985*   4997*   5000*   5014*   5034*   5037*   5052*
                5067*   5080*   5083*   5114*   5117*   5132*   5147*   5161*   5165    5183*   5186*   5201*   5216*
                5231*   5251*   5254*   5269*   5284*   5298*   5303    5321*   5324*   5337*   5340*   5355*   5368*
                5371*   5376    5419*   5422    5475*   5478    5480*   5483    5540*   5543
R1      =%000001  737*    739*    4076*   4077*   4078*   4088    4105*   4109    4124*   4127*   4131    4143*   4147
                4160*   4163    4177*   4181    4192*   4200    5423*   5427    5437*
R2      =%000002  482*    493*    500*    501     511*    512*    520*    521*    531*    532*    540*    541*    562*
                563     573*    574*    581*    582     592*    593*    600*    601     611*    612*    619*    620
                629*    630*    661*    662     672*    673*    680*    681     690*    691*    698*    699     708*
                709*    716*    717     726*    727*    734*    735     746*    747*    754*    755     757*    759*
                766*    767*    768*    770*    774*    775     786*    787*    794*    795     806*    807*    814*
                815     826*    827*    834*    835     846*    847*    870*    871     881*    882*    889*    890
                899*    900*    907*    908     917*    918*    925*    926     935*    936*    958*    959     970*
                971*    978*    979     989*    990*    999*    1000*   1007*   1008    1019*   1020*   1027*   1028
                1039*   1040*   1047*   1048    1059*   1060*   1081*   1082    1094*   1095*   1107*   1108*   1128*
                1129    1141*   1142*   1154*   1155*   1175*   1176    1188*   1189*   1201*   1202*   1222*   1223
                1235*   1236*   1248*   1249*   1291*   1292    1300*   1301*   1310*   1311*   1319*   1320*   1337*
                1338    1347*   1348*   1364*   1365*   1378*   1379    1387*   1388*   1399*   1400*   1415*   1416
                1425*   1426*   1438*   1439*   1453*   1454    1465*   1466*   1479*   1480*   1497*   1498    1510*
                1511*   1525*   1526*   1543*   1544    1557*   1558*   1573*   1574*   1591*   1592    1605*   1606*
                1620*   1621*   1633*   1634    1648*   1649*   1665*   1666*   1687*   1688    1700*   1701*   1714*
                1715*   1734*   1735    1749*   1750*   1767*   1768*   1788*   1789    1802*   1803*   1819*   1820*
                1827*   1828    1839*   1840*   1855*   1856*   1878*   1879    1890*   1891*   1904*   1905*   1920*
                1921    1932*   1933*   1944*   1945*   1961*   1962    1973*   1974*   1985*   1986*   2001*   2002
                2011*   2012*   2024*   2025*   2041*   2042    2057*   2058*   2073*   2074    2089*   2090*   2105*
                2106    2122*   2123*   2137*   2138    2154*   2155*   2171*   2172*   2186*   2187    2203*   2204*
                2213*   2214*   2229*   2230    2246*   2247*   2255*   2256*   2271*   2272*   2281*   2282*   2296*
                2297    2315*   2316*   2325*   2326*   2342*   2343    2362*   2363*   2377*   2378*   2387*   2388*
                2402*   2403    2419*   2420*   2428*   2429*   2443*   2444    2462*   2463*   2472*   2473*   2487*
                2488    2505*   2506*   2514*   2515*   2529*   2530    2547*   2548*   2556*   2557*   2570*   2571
                2583*   2584*   2597*   2598    2610*   2611*   2626*   2627    2636*   2637*   2649*   2650*   2660*
                2661*   2675*   2676*   2686*   2687*   2700*   2701    2712*   2713*   2721*   2722*   2731*   2732*
                2743*   2744*   2753*   2754*   2768*   2769    2782*   2783*   2796*   2797    2810*   2811*   2825*
                2826    2839*   2840*   2858*   2859    2874*   2875*   2884*   2885*   2899*   2900    2914*   2915*
                2929*   2930    2942*   2943*   2952*   2953*   2969*   2970    2982*   2983*   2991*   2992*   3005*
                3006    3020*   3021*   3030*   3031*   3046*   3047    3058*   3059*   3073*   3074    3085*   3086*
                3100*   3101    3112*   3113*   3132*   3133    3147*   3148*   3170*   3171    3185*   3186*   3208*
                3209    3222*   3223*   3240*   3241    3254*   3255*   3270*   3271    3285*   3286*   3300*   3301*
                3317*   3318    3333*   3334*   3348*   3349*   3365*   3366*   3380*   3381    3398*   3399*   3415*
                3416*   3434*   3435*   3449*   3450    3463*   3464*   3477*   3478*   3491*   3492*   3507*   3508
                3524*   3525*   3542*   3543    3560*   3561*   3576*   3577    3590*   3591*   3605*   3606    3620*
                3621*   3637*   3638    3647*   3648*   3656*   3657*   3671*   3672    3683*   3684*   3699*   3700
                3711*   3712*   3720*   3721*   3736*   3737    3747*   3748*   3763*   3764    3775*   3776*   3784*
                3785*   3802*   3803    3814*   3815*   3824*   3825*   3844*   3845    3857*   3858*   3877*   3878
```

		3890*	3891*	3932*	3933	3944*	3945*	3953*	3954*	3967*	3968*	3979*	3980*	3988*
		3989*	4001*	4002*	4014*	4015*	4028*	4029*	4042*	4043*	4067*	4068	4082*	4083*
		4101*	4102*	4120*	4121*	4138*	4139*	4156*	4157*	4172*	4173*	4188*	4189*	4207*
		4208*	4223*	4224	4232*	4233*	4241*	4242*	4263*	4264	4277*	4278*	4292*	4293*
		4300*	4301	4315*	4316*	4330*	4331*	4338*	4339	4353*	4354*	4367*	4368*	4375*
		4376	4391*	4392*	4406*	4407*	4426*	4427	4441*	4442*	4457*	4458*	4472*	4473*
		4480*	4481	4495*	4496*	4509*	4510*	4523*	4524*	4539*	4540*	4557*	4558	4572*
		4573*	4580*	4581	4595*	4596*	4610*	4611*	4618*	4619	4633*	4634*	4648*	4649*
		4666*	4667	4680*	4681*	4694*	4695*	4708*	4709*	4720*	4721*	4735*	4736*	4743*
		4744	4758*	4759*	4773*	4774*	4788*	4789*	4806*	4807	4822*	4823*	4838*	4839*
		4854*	4855*	4862*	4863	4877*	4878*	4892*	4893*	4908*	4909*	4922*	4923*	4930*
		4931	4945*	4946*	4964*	4965	4979*	4980*	4994*	4995*	5009*	5010*	5024*	5025*
		5031*	5032	5047*	5048*	5062*	5063*	5077*	5078*	5092*	5093*	5111*	5112	5127*
		5128*	5142*	5143*	5156*	5157*	5172*	5173*	5180*	5181	5196*	5197*	5211*	5212*
		5226*	5227*	5240*	5241*	5248*	5249	5264*	5265*	5279*	5280*	5293*	5294*	5310*
		5311*	5318*	5319	5334*	5335*	5350*	5351*	5365*	5366*	5383*	5384*	5416*	5417
		5447*	5448*	5449*	5465*	5466	5489*	5490*						
R3	=%000003	777*	779*	5425*	5452*									
R4	=%000004	797*	799*	1737*	1738*	1739*	1743*	1752*	1753*	1754*	1756*	1758*	1759*	1760*
		2575*	2576*	2577*	2601*	2603*	2604*	2630*	2641*	2642*	2643*	2652*	2653*	2654*
		2665*	2666*	2667*	2668*	2669*	2678*	2679*	2680*	2704*	2705*	2706	2715	2725
		2736*	2737	2746*	2747	2773*	2774*	2775*	2776	2802*	2803*	2804*	2831*	2832*
		2833	2865*	2866*	2867*	2868	2878*	2904*	2905*	2906*	2907	2935*	2936*	3009*
		3013*	3014*	5420*	5423									
R5	=%000005	817*	819*	5428*	5429*	5430	5453*	5454*	5455	5458*	5459*	5460		
R6	=%000006	405*	481*	492*	837*	839*	873*	4073*	4090	4092	4111	4113	4123*	4226*
		4227*	5537*											
R7	=%000007	404*	2004*											
SBC1	013756	5038	5039	5040	5046*									
SBC2	013766	5041	5050*											
SBC3	014004	5053	5054	5055	5061*									
SBC4	014014	5056	5065*											
SBC5	014032	5068	5069	5070	5076*									
SBC6	014042	5071	5080*											
SBC7	014062	5084	5085	5091*										
SB0	010066	3642	3650*											
SB2	010210	3706	3714*											
SB4	010334	3770	3778*											
SB5	010422	3804	3823*											
SB5A	010414	3809	3817*											
SB5X	010432	3806*	3808	3828*	3829									
SB5XAD	010434	3805	3817	3829*										
SB6	010474	3846	3856*											
SB6X	010504	3847*	3848	3861*										
SB7	010544	3879	3889*											
SB7X	010554	3880*	3894*	3895										
SB7XAD	010556	3881	3895*											
SCOPE =	000240	403*												
SETBR	014760	5422*	5461											
SETCC	015000	5426*	5456											
SETUP	014742	5419*												
SET2BR	015050	5431	5435*											
SHL	002014	1012*	1013											
SHR	002120	1052*	1053											
SNMB0A	004172	2081	2082	2088*										
SNMB1A	004276	2146	2147	2153*										

SNMB1B	004306	2148	2157*		
SNMB1C	004330	2162	2163	2164	2170*
SNMB2A	004452	2238	2239	2245*	
SNMB2B	004462	2240	2249*		
SNMB2C	004476	2250	2258*		
SNMB2D	004516	2262	2263	2264	2270*
SNMB2E	004526	2265	2274*		
SNMB3A	004670	2354	2355	2361*	
SNMB3B	004700	2356	2365*		
SNMB3C	004716	2368	2369	2370	2376*
SNMB3D	004726	2371	2380*		
SNM0A	004132	2048	2049	2050	2056*
SNM1A	004234	2113	2114	2115	2121*
SNM2A	004372	2194	2195	2196	2202*
SNM2B	004402	2197	2206*		
SNM3A	004602	2306	2307	2308	2314*
SNM3B	004612	2309	2318*		
SNM4A	004776	2411	2412	2418*	
SNM4B	005006	2413	2422*		
SNM5A	005060	2454	2455	2461*	
SNM5B	005070	2456	2465*		
SNM6A	005144	2497	2498	2504*	
SNM6B	005154	2499	2508*		
SNM7A	005226	2539	2540	2546*	
SNM7B	005236	2541	2550*		
SOPA	004050	2006	2014*		
SOPB	004070	2003	2016	2023*	
SOPB0A	002562	1382	1390*		
SOPB0B	002572	1391	1398*		
SOPB1A	002704	1460	1468*		
SOPB1B	002720	1469	1471	1478*	
SOPB1C	002764	1505	1513*		
SOPB1D	003002	1517	1524*		
SOPB2A	003136	1600	1608*		
SOPB2B	003154	1611	1619*		
SOPB2C	003224	1643	1651*		
SOPB2D	003246	1656	1664*		
SOPB3A	003376	1744	1752*		
SOPB3B	003422	1755	1757	1766*	
SOPB3C	003470	1797	1805*		
SOPB3D	003512	1811	1818*		
SOP>	004034	2005*	2014*	2015*	2028
SOP>A9	004100	2017*	2028*		
SOPZA	003050	1552	1560*		
SOP0A	002424	1295	1303*		
SOP0B	002442	1305	1313*		
SOP0C	002504	1342	1350*		
SOP0D	002526	1352	1355	1363*	
SOP1A	002630	1420	1428*		
SOP1B	002642	1429	1437*		
SOP2B	003070	1563	1572*		
SOP3A	003312	1695	1703*		
SOP3B	003326	1706	1713*		
SOP4A	003554	1834	1842*		
SOP4B	003574	1845	1854*		
SOP5A	003636	1885	1893*		

```
SOP5B   003652          1896    1903#
SOP6A   003716          1927    1935#
SOP6B   003732          1936    1943#
SOP7A   003776          1968    1976#
SOP7B   004012          1977    1984#
START   000500           479     489#  5545
STBOT = 000500           400#    481    492    873   4073   4090   4111   4123   4226   5537
SUB1    013604          4971    4972   4978#
SUB2    013614          4973    4982#
SUB3    013636          4986    4987   4993#
SUB4    013646          4988    4997#
SUB5    013664          5001    5002   5008#
SUB6    013674          5003    5012#
SUB7    013714          5015    5016   5017   5023#
SWB1    012646          4625    4626   4632#
SWB2    012656          4627    4636#
SWB3    012674          4639    4640   4641   4647#
TBL1    006536          3135    3136*  3137*  3138*  3139*  3140   3156#  3174*  3175*  3176*  3177*  3178   3189
                        3190    3191   3192   3193   3211   3212*  3213*  3214*  3215*  3216   3244*  3245*  3246*
                        3247*   3248
TBL2    006622          3173    3193#  3243
TEST1   012560          4586    4587   4588   4594#
TEST2   012570          4589    4598#
TEST3   012606          4602    4603   4609#
TPB   = 177566           408#   5478*  5543*
TPS   = 177564           407#   5476   5541
TST1    000554           500#
TST10   001114           682     685    698#
TST100  005724          2798    2805   2825#
TST101  005762          2827    2834   2858#
TST102  006042          2860    2879   2899#
TST103  006102          2901    2909   2929#
TST104  006154          2931    2947   2965-
TST105  006230          2971    2986   3005#
TST106  006306          3007    3025   3046#
TST107  006350          3048    3053   3073#
TST11   001146           700     703    716#
TST110  006412          3075    3080   3100#
TST111  006454          3102    3107   3132#
TST112  006540          3141    3170#
TST113  006624          3179    3208#
TST114  006710          3210    3217   3240#
TST115  006774          3242    3249   3270#
TST116  007070          3272    3294   3317#
TST117  007232          3319    3359   3380#
TST12   001200           718     721    734#
TST120  007410          3382    3428   3449#
TST121  007552          3451    3485   3507#
TST122  007626          3509    3518   3542#
TST123  007712          3554    3576#
TST124  007760          3578    3584   3605#
TST125  010036          3614    3637#
TST126  010104          3639    3651   3671#
TST127  010146          3673    3678   3699#
TST13   001234           736     741    754#
TST130  010226          3701    3715   3736#
```

TST131	010270	3738	3742	3763#	
TST132	010350	3765	3779	3802#	
TST133	010436	3818	3844#		
TST134	010506	3851	3877#		
TST135	010560	3884	3932#		
TST136	011124	4037	4067#		
TST137	011602	4201	4223#		
TST14	001300	774#			
TST140	011656	4225	4236	4263#	
TST141	011742	4265	4286	4300#	
TST142	012032	4302	4324	4338#	
TST143	012122	4340	4361	4375#	
TST144	012214	4377	4400	4426#	
TST145	012332	4428	4466	4480#	
TST146	012474	4482	4533	4557#	
TST147	012532	4559	4566	4580#	
TST15	001334	776	781	794#	
TST150	012616	4582	4604	4618#	
TST151	012704	4620	4642	4666#	
TST152	013070	4668	4729	4743#	
TST153	013206	4745	4782	4806#	
TST154	013332	4808	4848	4862#	
TST155	013512	4864	4916	4930#	
TST156	013552	4932	4939	4964#	
TST157	013724	4966	5018	5031#	
TST16	001370	796	801	814#	
TST160	014072	5033	5086	5111#	
TST161	014240	5113	5166	5180#	
TST162	014402	5182	5234	5248#	
TST163	014552	5250	5304	5318#	
TST164	014732	5320	5377	5416#	
TST165	015172	5465#			
TST17	001424	816	821	834#	
TST2	000644	502	535	562#	
TST20	001460	836	841	870#	
TST21	001520	872	876	889#	
TST22	001556	891	894	907#	
TST23	001614	909	912	925#	
TST24	001652	927	930	958#	
TST25	001716	960	965	978#	
TST26	001774	980	984	994	1007#
TST27	002034	1009	1014	1027#	
TST3	000700	564	568	581#	
TST30	002100	1029	1034	1047#	
TST31	002140	1049	1054	1081#	
TST32	002210	1083	1101	1128#	
TST33	002260	1130	1148	1175#	
TST34	002330	1177	1195	1222#	
TST35	002400	1224	1242	1291#	
TST36	002456	1293	1314	1337#	
TST37	002536	1339	1358	1378#	
TST4	000736	583	587	600#	
TST40	002602	1380	1393	1415#	
TST41	002652	1417	1432	1453#	
TST42	002730	1455	1473	1497#	
TST43	003012	1499	1519	1543#	

```
TST44   003100     1545   1367   1591#
TST45   003164     1593   1614   1633#
TST46   003256     1635   1659   1687#
TST47   003336     1689   1708   1734#
TST5    000774      602    606    619#
TST50   003432     1736   1761   1788#
TST51   003522     1790   1813   1827#
TST52   003604     1829   1849   1878#
TST53   003662     1880   1898   1920#
TST54   003742     1922   1938   1961#
TST55   004022     1963   1979   2001#
TST56   004102     2018   2041#
TST57   004142     2043   2051   2073#
TST6    001032      621    624    661#
TST60   004202     2075   2093   2105#
TST61   004244     2107   2116   2137#
TST62   004340     2139   2165   2186#
TST63   004420     2188   2208   2229#
TST64   004544     2231   2276   2296#
TST65   004630     2298   2320   2342#
TST66   004744     2344   2382   2402#
TST67   005022     2404   2423   2443#
TST7    001062      663    667    680#
TST70   005106     2445   2467   2487#
TST71   005170     2489   2509   2529#
TST72   005252     2531   2551   2570#
TST73   005306     2572   2578   2597#
TST74   005342     2599   2605   2626#
TST75   005504     2628   2681   2700#
TST76   005630     2702   2748   2768#
TST77   005666     2770   2777   2796#
WAIT    015240     5476#  5477   5479
WATE    015464     5541#  5542   5544
YBR     015104     5433#  5436#  5446#
YNTAB   015370     5420   5515#
$APTHD  000330      457    463#
$CPUOP  000326      439#
$DEVCT  000310      430#
$ENDAD  015264      416   5473   5483#
$ENV    000320      435#
$ENVM   000321      436#  5471   5538
$ERN  = 000353      398#   511    512#   520    521#   531    532#   540    541#   573    574#   592    593#
                    611    612#   629    630#   672    673#   690    691#   708    709#   726    727#   746
                    747#   767    768#   786    787#   806    807#   826    827#   846    847#   881    882#
                    899    900#   917    918#   935    936#   970    971#   989    990#   999   1000#  1019
                   1020#  1039   1040#  1059   1060#  1094   1095#  1107   1108#  1141   1142#  1154   1155#
                   1188   1189#  1201   1202#  1235   1236#  1248   1249#  1300   1301#  1310   1311#  1319
                   1320#  1347   1348#  1364   1365#  1387   1388#  1399   1400#  1425   1426#  1438   1439#
                   1465   1466#  1479   1480#  1510   1511#  1525   1526#  1557   1558#  1573   1574#  1605
                   1606#  1620   1621#  1648   1649#  1665   1666#  1700   1701#  1714   1715#  1749   1750#
                   1767   1768#  1802   1803#  1819   1820#  1839   1840#  1855   1856#  1890   1891#  1904
                   1905#  1932   1933#  1944   1945#  1973   1974#  1985   1986#  2011   2012#  2024   2025#
                   2057   2058#  2089   2090#  2122   2123#  2154   2155#  2171   2172#  2203   2204#  2213
                   2214#  2246   2247#  2255   2256#  2271   2272#  2281   2282#  2315   2316#  2325   2326#
                   2362   2363#  2377   2378#  2387   2388#  2419   2420#  2428   2429#  2462   2463#  2472
                   2473#  2505   2506#  2514   2515#  2547   2548#  2556   2557#  2583   2584#  2610   2611#
```

```
                                2636    2637#   2649    2650#   2660    2661#   2675    2676#   2686    2687#   2712    2713#   2721
                                2722#   2731    2732#   2743    2744#   2753    2754#   2782    2783#   2810    2811#   2839    2840#
                                2874    2875#   2884    2885#   2914    2915#   2942    2943#   2952    2953#   2982    2983#   2991
                                2992#   3020    3021#   3030    3031#   3058    3059#   3085    3086#   3112    3113#   3147    3148#
                                3185    3186#   3222    3223#   3254    3255#   3285    3286#   3300    3301#   3333    3334#   3349
                                3349#   3365    3366#   3398    3399#   3415    3416#   3434    3435#   3463    3464#   3477    3478#
                                3491    3492#   3524    3525#   3560    3561#   3590    3591#   3620    3621#   3647    3648#   3656
                                3657#   3683    3684#   3711    3712#   3720    3721#   3747    3748#   3775    3776#   3784    3785#
                                3814    3815#   3824    3825#   3857    3858#   3890    3891#   3944    3945#   3953    3954#   3967
                                3968#   3979    3980#   3988    3989#   4001    4002#   4014    4015#   4028    4029#   4042    4043#
                                4082    4083#   4101    4102#   4120    4121#   4138    4139#   4156    4157#   4172    4173#   4188
                                4189#   4207    4208#   4232    4233#   4241    4242#   4277    4278#   4292    4293#   4315    4316#
                                4330    4331#   4353    4354#   4367    4368#   4391    4392#   4406    4407#   4441    4442#   4457
                                4458#   4472    4473#   4495    4496#   4509    4510#   4523    4524#   4539    4540#   4572    4573#
                                4595    4596#   4610    4611#   4633    4634#   4648    4649#   4680    4681#   4694    4695#   4708
                                4709#   4720    4721#   4735    4736#   4758    4759#   4773    4774#   4788    4789#   4822    4823#
                                4838    4839#   4854    4855#   4877    4878#   4892    4893#   4908    4909#   4922    4923#   4945
                                4946#   4979    4980#   4994    4995#   5009    5010#   5024    5025#   5047    5048#   5062    5063#
                                5077    5078#   5092    5093#   5127    5128#   5142    5143#   5156    5157#   5172    5173#   5196
                                5197#   5211    5212#   5226    5227#   5240    5241#   5264    5265#   5279    5280#   5293    5294#
                                5310    5311#   5334    5335#   5350    5351#   5365    5366#   5383    5384#   5448    5449#   5489
                                5490#
$ERROR= 000302                   487#    495#
$ETABL  000320                   434#
$ETEND  000330                   446#    469
$FATAL  000302                   427#    487
$HIBTS  000330                   464#
$MAIL   000300                   425#    465     469
$MBAOR  000332                   465#
$MSGAD  000314                   432#
$MSGLG  000316                   433#
$MSGTY  000300                   426#    496#
$PASS   000306                   429#    490#    5468#
$PASTM  000336                   467#
$SVPC = 000400                   414#    419
$SWR  = 000000                   398#
$SWREG  000322                   437#
$TESTN  000304                   428#    482     488     493     766     770     5447
$TN   = 000166                   398#    497     503#    535     559     565#    568     578     584#    587     597     603#    606
                                 616     622#    624     658     664#    667     677     683#    685     695     701#    703     713
                                 719#    721     731     737#    741     751     757#    771     777#    781     791     797#    801
                                 811     817#    821     831     837#    841     867     873#    876     886     892#    894     904
                                 910#    912     922     928#    930     955     961#    965     975     981#    984     994     1004
                                1010#   1014    1024    1030#   1034    1044    1050#   1054    1078    1084#   1101    1125    1131#
                                1148    1172    1178#   1195    1219    1225#   1242    1288    1294#   1314    1334    1340#   1358
                                1375    1381#   1393    1412    1418#   1432    1450    1456#   1473    1494    1500#   1519    1540
                                1546#   1567    1588    1594#   1614    1630    1636#   1659    1684    1690#   1708    1731    1737#
                                1761    1785    1791#   1813    1824    1830#   1849    1875    1881#   1898    1917    1923#   1938
                                1958    1964#   1979    1998    2004#   2018    2038    2044#   2051    2070    2076#   2083    2102
                                2108#   2116    2134    2140#   2165    2183    2189#   2208    2226    2232#   2276    2293    2299#
                                2320    2339    2345#   2382    2399    2405#   2423    2440    2446#   2467    2484    2490#   2509
                                2526    2532#   2551    2567    2573#   2578    2594    2600#   2605    2623    2629#   2681    2697
                                2703#   2748    2765    2771#   2777    2793    2799#   2805    2822    2828#   2834    2855    2861#
                                2879    2896    2902#   2909    2926    2932#   2947    2966    2972#   2986    3002    3008#   3025
                                3043    3049#   3053    3070    3076#   3080    3097    3103#   3107    3129    3135#   3141    3167
                                3173#   3179    3205    3211#   3217    3237    3243#   3249    3267    3273#   3294    3314    3320#
```

```
$TSTM   000334
$TSTNM= 000304
$UNIT   000312
$UNITM  000340
$USWR   000324
$X      = 015202
$XX     = 177720

3359    3377    3383#   3428    3446    3452#   3485    3504    3510#   3518    3539    3545#   3554
3573    3579#   3584    3602    3608#   3614    3634    3640#   3651    3668    3674#   3678    3696
3702#   3715    3733    3739#   3742    3760    3766#   3779    3799    3805#   3818    3841    3847#
3851    3874    3880#   3884    3929    3935#   4037    4064    4070#   4201    4220    4226#   4236
4260    4266#   4286    4297    4303#   4324    4335    4341#   4361    4372    4378#   4400    4423
4429#   4466    4477    4483#   4533    4554    4560#   4566    4577    4593#   4604    4615    4621#
4642    4663    4669#   4729    4740    4746#   4782    4803    4809#   4848    4859    4665#   4916
4927    4933#   4939    4961    4967#   5018    5028    5034#   5086    5108    5114#   5166    5177
5183#   5234    5245    5251#   5304    5315    5321#   5377    5413    5419#   5462    5468#
 466#
 488#   494#
 431#
 468#
 438#
 503#   518     538     565#    571     584#    590     603#    609     622#    627     664#    670
 683#   688     701#    706     719#    724     737#    744     757#    777#    784     797#    804
 817#   824     837#    844     873#    879     892#    897     910#    915     928#    933     961#
 968    981#    987     997     1010#   1017    1030#   1037    1050#   1057    1084#   1091    1104
1131#   1138    1151    1178#   1185    1198    1225#   1232    1245    1294#   1298    1308    1317
1340#   1345    1361    1381#   1385    1396    1418#   1423    1435    1456#   1463    1476    1500#
1508    1522    1546#   1555    1570    1594#   1603    1617    1636#   1646    1662    1690#   1698
1711    1737#   1747    1764    1791#   1800    1816    1830#   1837    1852    1881#   1888    1901
1923#   1930    1941    1964#   1971    1982    2004#   2009    2021    2044#   2054    2076#   2086
2108#   2119    2140#   2151    2168    2189#   2200    2211    2232#   2243    2253    2268    2279
2299#   2312    2323    2345#   2359    2374    2385    2405#   2416    2426    2446#   2459    2470
2490#   2502    2512    2532#   2544    2554    2573#   2581    2600#   2608    2629#   2634    2647
2658    2673    2684    2703#   2710    2719    2729    2741    2751    2771#   2780    2799#   2808
2828#   2837    2861#   2872    2882    2902#   2912    2932#   2940    2950    2972#   2980    2989
3008#   3018    3028    3049#   3056    3076#   3083    3103#   3110    3135#   3144    3173#   3182
3211#   3220    3243#   3252    3273#   3282    3297    3320#   3330    3345    3362    3383#   3395
3412    3431    3452#   3460    3474    3488    3510#   3521    3545#   3557    3579#   3587    3608#
3617    3640#   3645    3654    3674#   3681    3702#   3709    3718    3739#   3745    3766#   3773
3782    3805#   3812    3821    3847#   3854    3880#   3887    3935#   3942    3951    3965    3977
3986    3999    4012    4026    4040    4070#   4098    4117    4135    4153    4169    4185    4204
4226#   4239    4266#   4274    4289    4303#   4312    4327    4341#   4350    4364    4378#   4388
4403    4429#   4438    4454    4469    4483#   4492    4506    4520    4536    4560#   4569    4583#
4592    4607    4621#   4630    4645    4669#   4677    4691    4705    4717    4732    4746#   4755
4770    4785    4809#   4819    4835    4851    4865#   4874    4889    4905    4919    4933#   4942
4967#   4976    4991    5006    5021    5034#   5044    5059    5074    5089    5114#   5124    5139
5153    5169    5183#   5193    5208    5223    5237    5251#   5261    5276    5290    5307    5321#
5331    5347    5362    5380    5419#   5468#
 518#    538#    571#    590#    609#    627#    670#    688#    706#    724#    744#    784#    804#
 824#    844#    879#    897#    915#    933#    968#    987#    997#   1017#   1037#   1057#   1091#
1104#   1138#   1151#   1185#   1198#   1232#   1245#   1298#   1308#   1317#   1345#   1361#   1385#
1396#   1423#   1435#   1463#   1476#   1508#   1522#   1555#   1570#   1603#   1617#   1646#   1662#
1698#   1711#   1747#   1764#   1800#   1816#   1837#   1852#   1888#   1901#   1930#   1941#   1971#
1982#   2009#   2021#   2054#   2086#   2119#   2151#   2168#   2200#   2211#   2243#   2253#   2268#
2279#   2312#   2323#   2359#   2374#   2385#   2416#   2426#   2459#   2470#   2502#   2512#   2544#
2554#   2581#   2608#   2634#   2647#   2658#   2673#   2684#   2710#   2719#   2729#   2741#   2751#
2780#   2808#   2837#   2872#   2882#   2912#   2940#   2950#   2980#   2989#   3018#   3028#   3056#
3083#   3110#   3144#   3182#   3220#   3252#   3282#   3297#   3330#   3345#   3362#   3395#   3412#
3431#   3460#   3474#   3488#   3521#   3557#   3587#   3617#   3645#   3654#   3681#   3709#   3718#
3745#   3773#   3782#   3812#   3821#   3854#   3887#   3942#   3951#   3965#   3977#   3986#   3999#
4012#   4026#   4040#   4098#   4117#   4135#   4153#   4169#   4185#   4204#   4239#   4274#   4289#
4312#   4327#   4350#   4364#   4388#   4403#   4438#   4454#   4469#   4492#   4506#   4520#   4536#
4569#   4592#   4607#   4630#   4645#   4677#   4691#   4705#   4717#   4732#   4755#   4770#   4785#
```

```
                    4819#  4835#  4851#  4874#  4889#  4905#  4919#  4942#  4976#  4991#  5006#  5021#  5044#
                    5059#  5074#  5089#  5124#  5139#  5153#  5169#  5193#  5208#  5223#  5237#  5261#  5276#
                    5290#  5307#  5331#  5347#  5362#  5380#
$XXX  = 000720       518#   538#   571#   590#   609#   627#   670#   688#   706#   724#   744#   784#   804#
                     824#   844#   879#   897#   915#   933#   968#   987#   997#  1017#  1037#  1057#  1091#
                    1104#  1138#  1151#  1185#  1198#  1232#  1245#  1298#  1308#  1317#  1345#  1361#  1385#
                    1396#  1423#  1435#  1463#  1476#  1508#  1522#  1555#  1570#  1603#  1617#  1646#  1662#
                    1698#  1711#  1747#  1764#  1800#  1816#  1837#  1852#  1888#  1901#  1930#  1941#  1971#
                    1982#  2009#  2021#  2054#  2086#  2119#  2151#  2168#  2200#  2211#  2243#  2253#  2268#
                    2279#  2312#  2323#  2359#  2374#  2385#  2416#  2426#  2459#  2470#  2502#  2512#  2544#
                    2554#  2581#  2608#  2634#  2647#  2658#  2673#  2684#  2710#  2719#  2729#  2741#  2751#
                    2780#  2808#  2837#  2872#  2882#  2912#  2940#  2950#  2980#  2989#  3018#  3028#  3056#
                    3083#  3110#  3144#  3182#  3220#  3252#  3282#  3297#  3330#  3345#  3362#  3395#  3412#
                    3431#  3460#  3474#  3488#  3521#  3557#  3587#  3617#  3645#  3654#  3681#  3709#  3718#
                    3745#  3773#  3782#  3812#  3821#  3854#  3887#  3942#  3951#  3965#  3977#  3986#  3999#
                    4012#  4026#  4040#  4098#  4117#  4135#  4153#  4169#  4185#  4204#  4239#  4274#  4289#
                    4312#  4327#  4350#  4364#  4388#  4403#  4438#  4454#  4469#  4492#  4506#  4520#  4536#
                    4569#  4592#  4607#  4630#  4645#  4677#  4691#  4705#  4717#  4732#  4755#  4770#  4785#
                    4819#  4835#  4851#  4874#  4889#  4905#  4919#  4942#  4976#  4991#  5006#  5021#  5044#
                    5059#  5074#  5089#  5124#  5139#  5153#  5169#  5193#  5208#  5223#  5237#  5261#  5276#
                    5290#  5307#  5331#  5347#  5362#  5380#
- 0.9526              409#   414    415#   417#   419#   420#   453    454#   456#   458#   470#   474#   477
                     478#   486#   503    518    538    565    571    584    590    603    609    622    627
                     664    670    683    688    701    706    719    724    737    744    757    777    784
                     797    804    817    824    837    844    873    879    892    897    910    915    928
                     933    961    968    981    987    997   1010   1017   1030   1037   1050   1057   1084
                    1091   1104   1131   1138   1151   1178   1185   1198   1225   1232   1245   1294   1298
                    1308   1317   1340   1345   1361   1381   1385   1396   1418   1423   1435   1456   1463
                    1476   1500   1508   1522   1546   1555   1570   1594   1603   1617   1636   1646   1662
                    1690   1698   1711   1737   1747   1764   1791   1800   1816   1830   1837   1852   1881
                    1888   1901   1923   1930   1941   1964   1971   1982   2004   2009   2021   2044   2054
                    2076   2086   2108   2119   2140   2151   2168   2189   2200   2211   2232   2243   2253
                    2268   2279   2299   2312   2323   2345   2359   2374   2385   2405   2416   2426   2446
                    2459   2470   2490   2502   2512   2532   2544   2554   2573   2581   2600   2608   2629
                    2634   2647   2658   2673   2684   2703   2710   2719   2729   2741   2751   2771   2780
                    2799   2808   2828   2837   2861   2872   2882   2902   2912   2932   2940   2950   2972
                    2980   2989   3008   3018   3028   3049   3056   3076   3083   3103   3110   3135   3144
                    3173   3182   3211   3220   3243   3252   3273   3282   3297   3320   3330   3345   3362
                    3383   3395   3412   3431   3452   3460   3474   3488   3510   3521   3545   3557   3579
                    3587   3608   3617   3640   3645   3654   3674   3681   3702   3709   3718   3739   3745
                    3766   3773   3782   3805   3812   3821   3847   3854   3880   3887   3935   3938   3942
                    3951   3965   3977   3986   3999   4012   4026   4040   4070   4098   4117   4135   4153
                    4169   4185   4204   4226   4239   4266   4274   4289   4303   4312   4327   4341   4350
                    4364   4378   4388   4403   4429   4438   4454   4469   4483   4492   4506   4520   4536
                    4560   4569   4583   4592   4607   4621   4630   4645   4669   4677   4691   4705   4717
                    4732   4746   4755   4770   4785   4809   4819   4835   4851   4865   4874   4889   4905
                    4919   4933   4942   4967   4976   4991   5006   5021   5034   5044   5059   5074   5089
                    5114   5124   5139   5153   5169   5183   5193   5208   5223   5237   5251   5261   5276
                    5290   5307   5321   5331   5347   5362   5380   5419   5430   5455   5460   5468   5496#
                    5498   5499   5500   5501   5502   5503   5504   5505   5506   5507   5508   5509   5510
                    5511   5512
.$.   = 000500       453#   458    477#   486
```

F11


```
COMMEN      1#
ENDCOM      1#
ERROR     398#   510   514   530   534   568   587   606   624   667   685   703   721   741   767
          781    801   821   841   876   894   912   930   965   984   994  1014  1034  1054  1088
         1101   1135  1148  1182  1195  1229  1242  1295  1305  1314  1342  1358  1382  1393  1420
         1432   1460  1473  1505  1519  1552  1567  1600  1614  1643  1659  1695  1708  1744  1761
         1797   1813  1834  1849  1885  1898  1927  1938  1968  1979  2006  2018  2051  2083  2116
         2148   2165  2197  2208  2240  2250  2265  2276  2309  2320  2356  2371  2382  2413  2423
         2456   2467  2499  2509  2541  2551  2576  2605  2631  2644  2655  2670  2681  2707  2716
         2726   2738  2748  2777  2805  2834  2869  2879  2909  2937  2947  2977  2986  3015  3025
         3053   3080  3107  3141  3179  3217  3249  3279  3294  3327  3342  3359  3392  3409  3429
         3457   3471  3485  3518  3554  3584  3614  3642  3651  3678  3706  3715  3742  3770  3779
         3809   3818  3851  3884  3939  3948  3962  3974  3983  3996  4009  4023  4037  4081  4095
         4114   4132  4150  4166  4182  4201  4232  4236  4271  4286  4309  4324  4347  4361  4385
         4400   4435  4451  4466  4489  4503  4517  4533  4566  4589  4604  4627  4642  4674  4688
         4702   4714  4729  4752  4767  4782  4816  4832  4848  4871  4886  4902  4916  4939  4973
         4988   5003  5018  5041  5056  5071  5086  5121  5136  5150  5166  5190  5205  5220  5234
         5258   5273  5287  5304  5328  5344  5359  5377  5448  5488
ESCAPE      1#
GETPRI      1#
GETSWR      1#
JNE      5387#  5428  5453  5458
LOOP      398#   518   538   571   590   609   627   670   688   706   724   744   784   804   824
          844    879   897   915   933   968   987   997  1017  1037  1057  1091  1104  1138  1151
         1185   1198  1232  1245  1298  1308  1317  1345  1361  1385  1396  1423  1435  1463  1476
         1508   1522  1555  1570  1603  1617  1646  1662  1698  1711  1747  1764  1800  1816  1837
         1852   1888  1901  1930  1941  1971  1982  2009  2021  2054  2086  2119  2151  2168  2200
         2211   2243  2253  2268  2279  2312  2323  2359  2374  2385  2416  2426  2459  2470  2502
         2512   2544  2554  2581  2608  2634  2647  2658  2673  2684  2710  2719  2729  2741  2751
         2780   2808  2837  2872  2882  2912  2940  2950  2980  2989  3018  3028  3056  3083  3110
         3144   3182  3220  3252  3282  3297  3330  3345  3362  3395  3412  3431  3460  3474  3488
         3521   3557  3587  3617  3645  3654  3681  3709  3718  3745  3773  3782  3812  3821  3854
         3887   3942  3951  3965  3977  3986  3999  4012  4026  4040  4098  4117  4135  4153  4169
         4185   4204  4239  4274  4289  4312  4327  4350  4364  4388  4403  4438  4454  4469  4492
         4506   4520  4536  4569  4592  4607  4630  4645  4677  4691  4705  4717  4732  4755  4770
         4795   4819  4835  4851  4874  4889  4905  4919  4942  4976  4991  5006  5021  5044  5059
         5074   5089  5124  5139  5153  5169  5193  5208  5223  5237  5261  5276  5290  5307  5331
         5347   5362  5380
MULT        1#
NEWTST      1#   398#  497   559   578   597   616   658   677   695   713   731   751   771   791
          911    831   867   886   904   922   955   975  1004  1024  1044  1078  1125  1172  1219
         1288   1334  1375  1412  1450  1494  1540  1588  1630  1684  1731  1785  1824  1875  1917
         1958   1998  2038  2070  2102  2134  2183  2226  2293  2339  2399  2440  2484  2526  2567
         2594   2623  2697  2765  2793  2822  2855  2896  2926  2966  3002  3043  3070  3097  3129
         3167   3205  3237  3267  3314  3377  3446  3504  3539  3573  3602  3634  3668  3696  3733
         3760   3799  3841  3874  3929  4064  4220  4260  4297  4335  4372  4423  4477  4554  4577
         4615   4663  4740  4803  4859  4927  4961  5028  5108  5177  5245  5315  5413  5462
POP         1#
PUSH        1#
REPORT      1#
SETPRI      1#
SETUP       1#
SKIP        1#
SLASH       1#
STARS       1#   398#  412   423   450   452   459   475   497   499   545   559   561   578   580
          597    599   616   618   634   658   660   677   679   695   697   713   715   731   733
```

	751	753	771	773	791	793	811	813	831	833	852	867	869	886	888
	904	906	922	924	940	955	957	975	977	1004	1006	1024	1026	1044	1046
	1065	1078	1080	1112	1125	1127	1159	1172	1174	1206	1219	1221	1253	1271	1275
	1288	1290	1325	1334	1336	1369	1375	1377	1404	1412	1414	1443	1450	1452	1485
	1494	1496	1530	1540	1542	1578	1588	1590	1625	1630	1632	1670	1684	1686	1719
	1731	1733	1772	1785	1787	1824	1826	1860	1875	1877	1909	1917	1919	1949	1958
	1960	1990	1998	2000	2030	2038	2040	2062	2070	2072	2094	2102	2104	2127	2134
	2136	2176	2183	2185	2218	2226	2228	2286	2293	2295	2330	2339	2341	2391	2399
	2401	2433	2440	2442	2477	2484	2486	2519	2526	2528	2561	2567	2569	2588	2594
	2596	2615	2623	2625	2691	2697	2699	2758	2765	2767	2787	2793	2795	2815	2822
	2824	2844	2855	2857	2889	2896	2898	2919	2926	2928	2957	2966	2968	2995	3002
	3004	3035	3043	3045	3063	3070	3072	3090	3097	3099	3117	3129	3131	3158	3167
	3169	3195	3205	3207	3227	3237	3239	3259	3267	3269	3305	3314	3316	3370	3377
	3379	3439	3446	3448	3496	3504	3506	3529	3539	3541	3566	3573	3575	3595	3602
	3604	3627	3634	3636	3661	3668	3670	3689	3696	3698	3726	3733	3735	3753	3760
	3762	3790	3799	3801	3832	3841	3843	3864	3874	3876	3898	3929	3931	4048	4064
	4066	4213	4220	4222	4246	4260	4262	4297	4299	4335	4337	4372	4374	4412	4423
	4425	4477	4479	4545	4554	4556	4577	4579	4615	4617	4653	4663	4665	4740	4742
	4793	4803	4805	4859	4861	4927	4929	4951	4961	4963	5028	5030	5098	5108	5110
	5177	5179	5245	5247	5315	5317	5390	5413	5415	5462	5464				
SWRSU	1#														
TYPBIN	1#														
TYPDEC	1#														
TYPNAM	1#														
TYPNUM	1#														
TYPOCS	1#														
TYPOCT	1#														
TYPTXT	1#														
SSERCD	398#	511	520	531	540	573	592	611	629	672	690	708	726	746	767
	786	806	826	846	881	899	917	935	970	989	999	1019	1039	1059	1094
	1107	1141	1154	1188	1201	1235	1248	1300	1310	1319	1347	1364	1387	1399	1425
	1438	1465	1479	1510	1525	1557	1573	1605	1620	1648	1665	1700	1714	1749	1767
	1802	1819	1839	1855	1890	1904	1932	1944	1973	1985	2011	2024	2057	2089	2122
	2154	2171	2203	2213	2246	2255	2271	2281	2315	2325	2362	2377	2387	2419	2428
	2462	2472	2505	2514	2547	2556	2583	2610	2636	2649	2660	2675	2686	2712	2721
	2731	2743	2753	2782	2810	2839	2874	2884	2914	2942	2952	2982	2991	3020	3030
	3058	3085	3112	3147	3185	3222	3254	3285	3300	3333	3348	3365	3398	3415	3434
	3463	3477	3491	3524	3560	3590	3620	3647	3656	3683	3711	3720	3747	3775	3784
	3814	3824	3857	3890	3944	3953	3967	3979	3988	4001	4014	4028	4042	4082	4101
	4120	4138	4156	4172	4188	4207	4232	4241	4277	4292	4315	4330	4353	4367	4391
	4406	4441	4457	4472	4495	4509	4523	4539	4572	4595	4610	4633	4648	4680	4694
	4708	4720	4735	4758	4773	4788	4822	4838	4854	4877	4892	4908	4922	4945	4979
	4994	5009	5024	5047	5062	5077	5092	5127	5142	5156	5172	5196	5211	5226	5240
	5264	5279	5293	5310	5334	5350	5365	5383	5448	5489					
$$ERMJ	398#	511	520	531	540	573	592	611	629	672	690	708	726	746	767
	786	806	826	846	881	899	917	935	970	989	999	1019	1039	1059	1094
	1107	1141	1154	1188	1201	1235	1248	1300	1310	1319	1347	1364	1387	1399	1425
	1438	1465	1479	1510	1525	1557	1573	1605	1620	1648	1665	1700	1714	1749	1767
	1802	1819	1839	1855	1890	1904	1932	1944	1973	1985	2011	2024	2057	2089	2122
	2154	2171	2203	2213	2246	2255	2271	2281	2315	2325	2362	2377	2387	2419	2428
	2462	2472	2505	2514	2547	2556	2583	2610	2636	2649	2660	2675	2686	2712	2721
	2731	2743	2753	2782	2810	2839	2874	2884	2914	2942	2952	2982	2991	3020	3030
	3058	3085	3112	3147	3185	3222	3254	3285	3300	3333	3348	3365	3398	3415	3434
	3463	3477	3491	3524	3560	3590	3620	3647	3656	3683	3711	3720	3747	3775	3784
	3814	3824	3857	3890	3944	3953	3967	3979	3988	4001	4014	4028	4042	4082	4101
	4120	4138	4156	4172	4188	4207	4232	4241	4277	4292	4315	4330	4353	4367	4391

CROSS REFERENCE TABLE -- MACRO NAMES

```
                4426    4441    4457    4472    4495    4509    4533    4539    4572    4595    4613    4633    4648    4680    4694
                4708    4720    4735    4758    4773    4798    4822    4838    4854    4877    4892    4908    4922    4945    4973
                4994    5009    5024    5047    5062    5077    5092    5127    5142    5156    5172    5196    5211    5225    5240
                5264    5279    5293    5310    5334    5350    5365    5383    5448    5489
$$ERRC           398#     535     568     587     606     624     667     685     703     721     741     781     801     821     841
                 876     894     912     930     965     984     994    1014    1034    1054    1101    1142    1135    1242    1314
                1358    1393    1432    1473    1519    1567    1614    1659    1708    1761    1813    1649    1898    1938    1979
                2018    2051    2083    2116    2165    2208    2276    2320    2382    2423    2467    2503    2551    2578    2605
                2681    2748    2777    2805    2834    2879    2909    2947    2986    3025    3053    3080    3107    3141    3179
                3217    3249    3294    3359    3428    3485    3518    3554    3584    3614    3651    3678    3715    3742    3779
                3818    3851    3894    4037    4201    4236    4286    4324    4361    4400    4466    4533    4566    4604    4642
                4729    4782    4848    4916    4939    5018    5086    5166    5234    5304    5377
$$ESCA             1#
$$LOCB           398#     518     538     571     590     609     627     670     688     706     724     744     784     804     824
                 844     879     897     915     933     968     987     997    1017    1037    1057    1091    1104    1138    1151
                1185    1198    1232    1245    1298    1308    1317    1345    1361    1385    1396    1423    1435    1463    1476
                1508    1522    1555    1570    1603    1617    1646    1662    1698    1711    1747    1764    1800    1816    1837
                1852    1888    1901    1930    1941    1971    1982    2009    2021    2054    2086    2119    2151    2168    2200
                2211    2243    2253    2268    2279    2312    2323    2359    2374    2385    2416    2426    2459    2470    2502
                2512    2544    2554    2581    2608    2634    2647    2658    2673    2684    2710    2719    2729    2741    2751
                2780    2808    2837    2872    2882    2912    2940    2950    2980    2989    3018    3028    3056    3083    3110
                3144    3182    3220    3252    3282    3297    3330    3345    3362    3395    3412    3431    3460    3474    3488
                3521    3557    3587    3617    3645    3654    3681    3709    3718    3745    3773    3782    3812    3821    3854
                3887    3942    3951    3965    3977    3986    3999    4012    4026    4040    4098    4117    4135    4153    4169
                4185    4204    4239    4274    4289    4312    4327    4350    4364    4388    4403    4438    4454    4469    4492
                4506    4520    4536    4569    4592    4607    4630    4645    4677    4691    4705    4717    4732    4755    4770
                4785    4819    4835    4851    4874    4889    4905    4919    4942    4976    4991    5006    5021    5044    5059
                5074    5089    5124    5139    5153    5169    5193    5208    5223    5237    5261    5276    5290    5307    5331
                5347    5362    5380
$$MEMT             1#     398#     497     559     578     597     616     658     677     695     713     731     751     771     791
                 811     831     867     886     904     922     955     975    1004    1024    1044    1078    1125    1172    1219
                1288    1334    1375    1412    1450    1494    1540    1588    1630    1684    1731    1785    1824    1875    1917
                1958    1998    2038    2070    2102    2134    2183    2226    2293    2339    2399    2440    2484    2526    2567
                2594    2623    2697    2765    2793    2822    2855    2896    2926    2966    3002    3043    3070    3097    3129
                3167    3205    3237    3267    3314    3377    3446    3504    3539    3573    3602    3634    3668    3696    3733
                3760    3799    3841    3874    3929    4064    4220    4260    4297    4335    4372    4423    4477    4554    4577
                4615    4663    4740    4803    4859    4927    4961    5028    5108    5177    5245    5315    5413    5462
$$SKIP             1#
.EQUAT             1#
.HEADE             1#
.KT11              1#
.SETUP             1#
.SWRHI             1#
.$ACTI             1#     409#     410
.$APTB             1#     409#     421
.$APTH             1#     409#     448
.$APTY             1#
.$ASTA             1#
.$CATC             1#
.$CMTA             1#
.$DB2D             1#
.$DB2O             1#
.$DIV              1#
.$EOP              1#
.$ERRC             1#
.$ERRT             1#
```

```
.$[illegible]     1#
.$POKE     1#
.$RAND     1#
.$RDDE     1#
.$RDOC     1#
.$READ     1#
.$R2AZ     1#
.$SAVE     1#
.$SB2D     1#
.$SB2O     1#
.$SCOP     1#
.$SIZE     1#
.$SUPR     1#
.$TRAP     1#
.$TYPB     1#
.$TYPD     1#
.$TYPE     1#
.$TYPO     1#
.$40CA     1#
.[illegible]     1#
```

```
.ABS    1351    1431    4749    4764    4778
.ABS    2576    2668    2776    3137    3175    3213    3245    4671    4685    4699    4715    4735
.ASC    5254    5269    5284    5298
.ASR    5324    5340    5355    5371
B00      740     760     780     800     820     840     984    1013    1053    1228    3277    3292    3340    3357    3387
        3469    3483    3550    3582    4397    4450    4727    4813    4830    4900    5016    5062    5118    5139    5188
        5203    5255    5270    5326    5342    5511
B01     1229    2050    2114    2163    2195    2263    2307    2369    3325    3405    3423    3455    3514    3612    4382
        4489    4533    4565    4588    4640    4780    4846    5038    5053    5299    5512
        504     535     568     587     606     624     667     685     703     721     741     761     781     801     821
        841     876     894     912     930     965     994    1014    1034    1054    1088    1100    1295    1314    1358
       1382    1393    1420    1432    1460    1473    1505    1519    1552    1567    1600    1614    1643    1659    1695
       1708    1744    1761    1797    1813    1834    1849    1885    1898    1927    1938    1968    1979    2006    2018
       2051    2116    2165    2197    2208    2250    2265    2276    2309    2320    2371    2423    2467    2509    2551
       2578    2605    2631    2644    2655    2670    2681    2726    2738    2777    2805    2834    2869    2909    2937
       2947    2977    2986    3015    3025    3053    3080    3107    3141    3179    3217    3249    3279    3294    3327
       3342    3359    3392    3409    3428    3457    3471    3485    3518    3554    3584    3614    3651    3678    3706
       3715    3742    3770    3779    3809    3818    3851    3884    3939    3948    3962    3974    3983    3996    4009
       4023    4037    4095    4114    4132    4150    4166    4182    4201    4236    4385    4399    4451    4487    4531
       4566    4589    4642    4729    4782    4815    4831    4901    5017    5041    5056    5070    5120    5135    5166
       5189    5204    5219    5257    5272    5300    5304    5327    5343    5358    5375    5377    5474    5481    5500
BGE    5501
BGT    2707    5503
BHI    4284    4322    4359    4501    4700    4765    4869    4937    4971    5507
BIC    2642    2667    4344    4358    4526    4825    4911    5080    5337    5429    5454    5459
BICB   2975    3138    3176    3214    3246
BIS    2652    3051    4381    4396    4444    4761    4895    4982    5368
BISB   2803    2907    3078    3105    3139    3177    3215    3247
BIT    2737    2747    4306    4321    5427
BITB   5471    5538
BLE    5504
BLOS   2082    2147    2239    2355    2412    2455    2498    2540    4269    4307    4345    4433    4464    4515    4602
       4625    4672    4686    4712    4750    4884    4914    4986    5001    5084    5148    5162    5217    5232    5285
       5356    5372    5508
BLT    2716    5502
BMI    1135    1147    1242    1305    1342    2049    2083    2115    2148    2164    2196    2240    2264    2308    2356
       2370    2382    2413    2456    2499    2541    2748    2879    3642    4271    4309    4347    4384    4400    4448
       4465    4486    4502    4517    4530    4563    4586    4604    4627    4641    4688    4714    4728    4752    4781
       4816    4832    4902    5018    5040    5055    5071    5121    5190    5234    5258    5328    5506
BNE     502     515     524     564     583     602     621     663     682     700     718     736     756     776     796
        816     836     872     891     909     927     960     980    1009    1029    1049    1083    1087    1101    1130
       1177    1224    1293    1339    1352    1380    1417    1455    1499    1545    1593    1635    1689    1736    1790
       1829    1880    1922    1963    2003    2043    2075    2107    2139    2188    2231    2298    2344    2404    2445
       2489    2531    2572    2599    2628    2702    2770    2798    2827    2860    2901    2931    2971    3007    3048
       3075    3102    3134    3172    3210    3242    3272    3319    3382    3389    3407    3425    3451    3509    3516
       3544    3552    3578    3607    3639    3673    3701    3738    3765    3804    3846    3879    3934    4069    4087
       4089    4091    4093    4108    4110    4112    4130    4146    4148    4162    4164    4180    4199    4225    4265
       4302    4340    4377    4428    4482    4559    4582    4620    4668    4745    4808    4847    4864    4932    4966
       5033    5113    5182    5250    5320    5418    5467    5470    5472    5479    5499    5539    5544
BPL    1134    1148    1355    1391    1429    1469    1471    1517    1563    1611    1656    1706    1755    1757    1811
       1845    1896    1936    1977    2016    4286    4324    4361    4434    4466    4674    4702    4767    4848    4871
       4886    4916    4939    4973    4988    5003    5086    5136    5150    5164    5205    5220    5273    5287    5302
       5344    5359    5374    5477    5505    5542
BP      505     525    4070    5498
BVC    1181    1195    3276    3291    3324    3339    3356    4503    4532    4687    4701    4751    4829    4885    4899
       4987    5015    5085    5134    5163    5202    5233    5271    5301    5341    5373    5509
```

```
B.S     1182  1194  1241  2048  2081  2113  2146  2162  2194  2238  2262  2306  2354  2368  24..
        2454  2497  2539  4270  4285  4308  4323  4346  4360  4383  4398  4435  4449  4488  45.6
        4564  4587  4603  4626  4639  4673  4713  4726  4766  4779  4814  4845  4870  4915  4938
        4972  5002  5039  5054  5069  5119  5149  5187  5218  5256  5286  5325  5357  5510
CCC      503  1085  1132  1179  1226  4430  4697  4826  4866  4896  4934  4968  5012  5115  5159
        5139  5229  5252  5296  5369
CLC      738   758   778   798   818   838   961  1011  1031  1051  3322  3385  3403  3467  3548
        3610  4267  4305  4343  4380  4462  4513  4724  4777  4811  5051  5131  5215  5268  5339
CLN     1146  2079  2144  2236  2352  2452  2495  2537  4267  4305  4343  4380  4395  4513  4528
        4600  4623  4748  4811  5066  5146  5185  5283  5323
CLR     1294  1340  1418  1419  1456  1457  1500  1501  1546  1549  1551  1594  1597  1636  1639
        1690  1693  1694  1737  1740  1741  1791  1794  1830  1833  1881  1882  1884  1923  1926
        1964  1967  2004  2044  2076  2108  2109  2140  2141  2157  2189  2190  2232  2233  2299
        2300  2345  2346  2405  2406  2446  2447  2490  2491  2532  2533  2573  2575  2600  2601
        2629  2663  2665  2703  2704  2734  2736  2771  2773  2774  2799  2800  2802  2828  2829
        2831  2861  2862  2865  2902  2903  2904  2932  2933  2972  3008  3009  3010  3077  3104
        3320  3352  3383  3401  3418  3675  3703  3935  4075  4076  4378  4562  4841
CLRB    1381  1459  1504  1599  1642  1743  1796
CLV     1193  1240  4528  4748  4882  4984  5082  5131  5268  5339
CLZ     1099  2046  2111  2160  2192  2260  2304  2366  2409  4282  4320  4357  4446  4499  4561
        4584  4637  4724  4763  4777  4843  5036  5051
CMP      501   563   582   586   601   605   620   623   662   681   684   699   702   717   720
         735   755   775   795   815   835   871   890   893   908   911   926   929   959   964
         979   993  1008  1028  1033  1048  1082  1129  1176  1223  1292  1338  1379  1416  1454
        1498  1544  1592  1634  1688  1735  1789  1828  1879  1921  1962  2002  2042  2074  2106
        2138  2187  2230  2297  2343  2403  2444  2488  2530  2571  2598  2627  2701  2706  2715
        2725  2769  2797  2826  2859  2900  2930  2970  3006  3047  3074  3079  3101  3106  3133
        3140  3171  3178  3209  3216  3241  3248  3271  3278  3293  3318  3326  3341  3358  3381
        3388  3406  3424  3450  3456  3470  3484  3508  3515  3543  3551  3577  3583  3606  3613
        3638  3650  3672  3677  3700  3705  3737  3741  3764  3769  3803  3808  3817  3845  3850
        3878  3883  3933  3938  3947  3973  3982  3995  4008  4022  4036  4068  4086  4088  4090
        4092  4094  4109  4111  4113  4129  4131  4145  4147  4149  4161  4163  4165  4179  4181
        4198  4200  4224  4235  4264  4301  4339  4376  4427  4481  4558  4581  4619  4667  4744
        4807  4863  4868  4883  4898  4913  4931  4965  5032  5112  5165  5181  5249  5303  5319
        5376  5417  5466  5473
CMPB    2833
COM     1313  1458  1502  1550  1562  1598  1640  1705  1742  1795  1844  1895  1935  1976  2407
        2448  2492  2534  2574  2602  2640  2641  2666  2735  2772  2801  2830  2864  2866  2877
        2934  2974  3011  4077  4936  4997
COMB    1390  1547  1595  1637  1691  1738  1792  1831  1924  1965  2077  2142  2234  2301  2319
        2347  2348  2449  2466  2493  2508  2535  2550  2664  2804  2832  2863  2906  2985  3012
DEC     1341  1357  1428  1513  1560  1561  1564  1565  1608  1612  1651  1652  1654  1657  1703
        1704  1752  1753  1758  1759  1805  1806  1807  1808  2206  2207  2249  2274  2275  2302
        2318  2643  2746  2945  2946  3390  3391  3408  3426  3427  4485  4500  4514  4529  4598
        5452  5457
HALT     409   513   522   533   542   575   594   613   631   674   692   710   728   748   769
         788   808   828   848   883   901   919   937   972   991  1001  1021  1041  1061  1096
        1109  1143  1156  1190  1203  1237  1250  1302  1312  1321  1349  1366  1389  1401  1427
        1440  1467  1481  1512  1527  1559  1575  1607  1622  1650  1667  1702  1716  1751  1769
        1804  1821  1841  1857  1892  1906  1934  1946  1975  1987  2013  2026  2059  2091  2124
        2156  2173  2205  2215  2248  2257  2273  2283  2317  2327  2364  2379  2389  2421  2430
        2464  2474  2507  2516  2549  2558  2585  2612  2638  2651  2662  2677  2688  2714  2723
        2733  2745  2755  2784  2812  2841  2876  2886  2916  2944  2954  2984  2993  3022  3032
        3060  3087  3114  3149  3187  3224  3256  3287  3302  3335  3350  3367  3400  3417  3436
        3465  3479  3493  3526  3562  3592  3622  3649  3658  3685  3713  3722  3749  3777  3786
        3816  3826  3859  3892  3946  3955  3969  3981  3990  4003  4016  4030  4044  4084  4103
```

	4122	4140	4158	4174	4190	4209	4234	4243	4279	4294	4317	4332	4355	4369	4393
	4408	4443	4459	4474	4497	4511	4525	4541	4574	4597	4612	4635	4650	4682	4696
	4710	4722	4737	4760	4775	4790	4824	4840	4856	4879	4894	4910	4924	4947	4981
	4996	5011	5026	5049	5064	5079	5094	5129	5144	5158	5174	5198	5213	5228	5242
	5266	5281	5295	5312	5336	5352	5367	5385	5450	5491	5534				
	500	512	521	532	541	562	574	581	593	600	612	619	630	661	673
	680	691	698	709	716	727	734	747	754	768	774	787	794	807	814
	827	834	847	870	882	889	900	907	918	925	936	958	971	978	990
	1000	1007	1020	1027	1040	1047	1060	1081	1095	1108	1128	1142	1155	1175	1183
	1202	1222	1236	1249	1291	1301	1303	1311	1320	1337	1348	1365	1378	1388	1400
	1415	1426	1439	1453	1466	1468	1480	1497	1503	1511	1514	1515	1526	1543	1548
	1558	1566	1574	1591	1596	1606	1609	1621	1633	1638	1641	1649	1653	1666	1687
	1692	1701	1707	1715	1734	1739	1750	1754	1768	1788	1793	1803	1809	1820	1827
	1832	1840	1842	1843	1846	1847	1848	1856	1878	1891	1893	1894	1897	1905	1920
	1925	1933	1937	1945	1961	1966	1974	1978	1986	2001	2012	2014	2017	2025	2041
	2058	2073	2090	2105	2123	2137	2155	2158	2172	2186	2204	2214	2229	2247	2256
	2258	2272	2282	2296	2316	2326	2342	2349	2363	2378	2388	2402	2420	2429	2443
	2450	2463	2465	2473	2487	2506	2515	2529	2548	2557	2570	2577	2584	2597	2604
	2611	2626	2637	2639	2650	2653	2654	2661	2669	2676	2680	2687	2700	2705	2713
	2722	2724	2732	2744	2754	2768	2775	2783	2796	2811	2825	2840	2858	2875	2885
	2899	2905	2908	2915	2925	2936	2943	2953	2969	2983	2992	3005	3014	3021	3031
	3046	3052	3059	3073	3086	3100	3113	3132	3148	3170	3186	3208	3223	3240	3255
	3270	3286	3301	3317	3334	3349	3353	3366	3380	3399	3416	3420	3435	3449	3464
	3478	3492	3507	3525	3542	3561	3576	3591	3605	3621	3637	3648	3657	3671	3684
	3699	3712	3721	3736	3748	3763	3776	3785	3802	3815	3825	3844	3858	3877	3891
	3932	3945	3954	3957	3968	3970	3980	3989	3992	4002	4005	4015	4018	4029	4032
	4043	4067	4083	4102	4104	4121	4125	4139	4141	4157	4159	4173	4175	4189	4191
	4208	4223	4233	4242	4263	4278	4293	4300	4316	4331	4338	4354	4368	4375	4392
	4407	4426	4432	4442	4447	4458	4463	4473	4480	4496	4510	4524	4540	4557	4573
	4580	4596	4611	4618	4634	4649	4666	4681	4695	4709	4721	4736	4743	4759	4774
	4789	4806	4823	4839	4855	4862	4878	4893	4909	4923	4930	4946	4964	4980	4995
	5010	5025	5031	5048	5063	5078	5093	5111	5128	5143	5157	5173	5180	5197	5212
	5227	5241	5248	5265	5280	5294	5311	5318	5335	5351	5366	5384	5416	5426	5449
	5465	5468	5490												
INCB	1392	1472	1518	1613	1658	1760	1812	5469							
JMP	479	483	3937	3958	3972	3993	4006	4019	4033	4071	5430	5431	5434	5443	5445
	5455	5456	5460	5461	5487	5545									
JSR	4078	4105	4127	4143	4160	4177	4192	5483							
MOV	481	482	489	490	491	492	493	494	495	496	511	520	531	540	565
	573	584	592	603	611	622	629	665	672	683	690	701	708	719	726
	737	746	757	766	767	770	777	786	797	806	817	826	837	846	873
	874	881	892	899	910	917	928	935	962	970	981	989	999	1010	1019
	1030	1039	1050	1059	1094	1107	1141	1154	1188	1201	1235	1248	1300	1310	1319
	1347	1364	1387	1399	1425	1438	1465	1479	1510	1525	1557	1573	1605	1620	1648
	1665	1700	1714	1749	1767	1802	1819	1839	1855	1890	1904	1932	1944	1973	1985
	2011	2024	2057	2089	2122	2154	2171	2203	2213	2246	2255	2271	2281	2315	2325
	2362	2377	2387	2419	2428	2462	2472	2505	2514	2547	2556	2583	2603	2610	2630
	2636	2649	2660	2675	2686	2712	2721	2731	2743	2753	2782	2810	2839	2874	2884
	2914	2935	2942	2952	2973	2982	2991	3020	3030	3049	3050	3058	3076	3085	3103
	3112	3135	3136	3147	3173	3174	3185	3211	3212	3222	3243	3244	3254	3273	3285
	3288	3300	3321	3333	3337	3348	3351	3365	3384	3398	3402	3415	3419	3434	3452
	3463	3466	3477	3480	3491	3510	3511	3524	3545	3546	3547	3560	3579	3590	3608
	3609	3620	3640	3647	3656	3674	3683	3702	3711	3720	3739	3747	3766	3767	3775
	3784	3805	3806	3814	3824	3847	3848	3857	3880	3881	3890	3936	3944	3953	3956
	3967	3971	3979	3988	3991	4001	4004	4014	4017	4028	4031	4042	4073	4074	4082
	4101	4120	4123	4124	4126	4138	4142	4156	4172	4176	4188	4207	4226	4227	4229

```
        4232  4241  4268  4277  4283  4292  4303  4315  4330  4341  4353  4367  4391  4406  4429
        4441  4457  4472  4483  4495  4509  4523  4539  4572  4595  4610  4621  4633  4648  4669
        4680  4694  4708  4720  4735  4746  4758  4773  4788  4809  4822  4838  4854  4865  4877
        4880  4892  4908  4922  4933  4945  4967  4979  4994  5009  5024  5034  5047  5062  5077
        5092  5114  5127  5142  5156  5172  5183  5196  5211  5226  5240  5251  5264  5279  5293
        5310  5321  5334  5350  5365  5383  5419  5420  5421  5422  5423  5424  5425  5428  5432
        5433  5435  5436  5439  5447  5448  5453  5458  5475  5480  5489  5533  5536  5537  5540
MOVB    2867  2878  3013  5478  5543
NEG     1304  1356  2015  4812  4828  4844
NEGB    1470  1516  1610  1655  1756  1810  1883  2679
NOP      403  5484  5485  5486
RESET   5482
ROL      739   759   779   799   819   839   963   983  1012  3275  3323  3386  3454  3513  3549
        3581  3611  5117  5132  5147  5161  5437
ROLB    3290  3338  3355  3404  3422  3468  3482
ROR     1032  1052  5186  5201  5216  5231
RTS     4229
SBC     1354  5037  5052  5067  5083
SCC     1098  1145  1192  1239  2045  2078  2110  2143  2159  2191  2235  2259  2303  2351  2365
        2408  2451  2494  2536  4266  4281  4304  4319  4342  4356  4379  4394  4461  4484  4512
        4527  4560  4583  4599  4622  4636  4670  4723  4747  4762  4776  4810  4842  4881  4912
        4983  4998  5035  5050  5065  5081  5130  5145  5184  5214  5267  5282  5322  5338  5353
SEC      982  1227  1350  1353  1430  3274  3289  3336  3354  3421  3453  3481  3512  3580  4445
        4498  4867  4935  4969  5160  5230  5253  5297  5370
SEN     1133  4698  4867  4969  5200  5253
SEV     1180  5116
SEZ      523  1086  4431  4683  4827  4897  4935  5013  5116  5160  5200  5230  5297  5370
SUB     2678  3714  4970  4985  5000  5014
SWAB    3641  3676  3704  3740  3768  3807  3849  3882  4624  4638
TST      567   666   875  2047  2112  2193  2305  2350  2380  2381  2410  2422  2453  2496  2538
        2868  3023  3024  3517  3553  3778  3961  4107  4585  4601
TSTB    2080  2145  2161  2237  2261  2353  2367  2976  5476  5541
.ASCIZ  5493  5547
.BYTE    435   436
.ENABL     1   399
.END    5551
.ENDC    417   419   448   514   523   534   544   577   596   615   633   676   694   712   730
         750   770   790   810   830   850   885   903   921   939   974   993  1003  1023  1043
        1063  1097  1111  1144  1158  1191  1205  1238  1252  1303  1313  1323  1350  1368  1390
        1403  1428  1442  1468  1483  1513  1529  1560  1577  1608  1624  1651  1669  1703  1718
        1752  1771  1805  1823  1842  1859  1893  1908  1935  1948  1976  1989  2014  2028  2061
        2093  2126  2157  2175  2206  2217  2249  2258  2274  2285  2318  2329  2365  2380  2391
        2422  2432  2465  2476  2508  2518  2550  2560  2587  2614  2639  2652  2663  2678  2690
        2715  2724  2734  2746  2757  2786  2814  2843  2877  2888  2918  2945  2956  2985  2995
        3023  3034  3062  3089  3116  3151  3189  3226  3258  3288  3304  3336  3351  3369  3401
        3418  3438  3466  3480  3495  3528  3564  3594  3624  3650  3660  3687  3714  3724  3751
        3778  3788  3817  3828  3861  3894  3947  3956  3970  3982  3991  4004  4017  4031  4046
        4085  4104  4123  4141  4159  4175  4191  4211  4235  4245  4280  4296  4318  4334  4356
        4371  4394  4410  4444  4460  4476  4498  4512  4526  4543  4576  4598  4614  4636  4651
        4683  4697  4711  4723  4739  4761  4776  4792  4825  4841  4858  4880  4895  4911  4926
        4949  4982  4997  5012  5027  5050  5065  5080  5096  5130  5145  5159  5176  5199  5214
        5229  5244  5267  5282  5296  5314  5337  5353  5368  5387  5451  5492
.EVEN     424  5[illegible]
.IF       415   417   446   448   511   515   523   531   535   568   587   606   624   667   685
         703   721   741   767   781   801   821   841   876   894   912   930   965   984   994
        1014  1034  1054  1088  1097  1101  1135  1144  1148  1182  1191  1195  1229  1238  1242
```

1295 1303 1305 1313 1314 1342 1350 1358 1382 1390 1393 1420 1428 1432 1460
1468 1473 1505 1513 1519 1552 1560 1567 1600 1608 1614 1643 1651 1659 1695
1703 1708 1744 1752 1761 1797 1805 1813 1834 1842 1849 1885 1893 1898 1927
1935 1938 1968 1976 1979 2006 2014 2018 2051 2083 2116 2148 2157 2165 2197
2206 2208 2240 2249 2250 2258 2265 2274 2276 2309 2318 2320 2356 2365 2371
2380 2382 2413 2422 2423 2456 2465 2467 2499 2508 2509 2541 2550 2551 2578
2605 2631 2639 2644 2652 2655 2663 2670 2678 2681 2707 2715 2716 2724 2726
2734 2738 2746 2748 2777 2805 2834 2869 2877 2879 2909 2937 2945 2947 2977
2985 2986 3015 3023 3025 3053 3080 3107 3141 3179 3217 3249 3279 3288 3294
3327 3336 3342 3351 3359 3392 3401 3409 3418 3428 3457 3466 3471 3480 3485
3518 3554 3584 3614 3642 3650 3651 3678 3706 3714 3715 3742 3770 3778 3779
3809 3817 3818 3851 3884 3939 3947 3948 3956 3962 3970 3974 3982 3983 3991
3996 4004 4009 4017 4023 4031 4037 4081 4095 4104 4114 4123 4132 4141 4150
4159 4166 4175 4182 4191 4201 4232 4236 4271 4280 4286 4309 4318 4324 4347
4356 4361 4385 4394 4400 4435 4444 4451 4460 4466 4489 4498 4503 4512 4517
4526 4533 4566 4589 4598 4604 4627 4636 4642 4674 4683 4688 4697 4702 4711
4714 4723 4729 4752 4761 4767 4776 4782 4816 4825 4832 4841 4848 4871 4880
4886 4895 4902 4911 4916 4939 4973 4982 4988 4997 5003 5012 5018 5041 5050
5056 5065 5071 5080 5086 5121 5130 5136 5145 5150 5159 5166 5190 5199 5205
5214 5220 5229 5234 5258 5267 5273 5282 5287 5296 5304 5328 5337 5344 5353
5359 5368 5377 5448 5488

.IFF
417 419 511 523 531 535 544 568 577 587 596 606 615 624 633
667 676 685 694 703 712 721 730 741 750 767 781 790 801 810
821 830 841 850 876 885 894 903 912 921 930 939 965 974 984
993 994 1003 1014 1023 1034 1043 1054 1063 1097 1101 1111 1144 1148 1158
1191 1195 1205 1238 1242 1252 1303 1313 1314 1323 1350 1358 1368 1390 1393
1403 1428 1432 1442 1468 1473 1483 1513 1519 1529 1560 1567 1577 1608 1614
1624 1651 1659 1669 1703 1708 1718 1752 1761 1771 1805 1813 1823 1842 1849
1859 1893 1898 1908 1935 1938 1948 1976 1979 1989 2014 2018 2028 2051 2061
2083 2093 2116 2126 2157 2165 2175 2206 2208 2217 2249 2258 2274 2276 2285
2318 2320 2329 2365 2380 2382 2391 2422 2423 2432 2465 2467 2476 2508 2509
2518 2550 2551 2560 2578 2587 2605 2614 2639 2652 2663 2678 2681 2690 2715
2724 2734 2746 2748 2757 2777 2786 2805 2814 2834 2843 2877 2879 2888 2909
2918 2945 2947 2956 2985 2986 2995 3023 3025 3034 3053 3062 3080 3089 3107
3116 3141 3151 3179 3189 3217 3226 3249 3258 3288 3294 3304 3336 3351 3359
3369 3401 3418 3428 3438 3466 3480 3485 3495 3518 3528 3554 3564 3584 3594
3614 3624 3650 3651 3660 3678 3687 3714 3715 3724 3742 3751 3778 3779 3788
3817 3818 3828 3851 3861 3884 3894 3947 3956 3970 3982 3991 4004 4017 4031
4037 4046 4081 4104 4123 4141 4159 4175 4191 4201 4211 4232 4236 4245 4280
4286 4296 4318 4324 4334 4356 4361 4371 4394 4400 4410 4444 4460 4466 4476
4498 4512 4526 4533 4543 4566 4576 4598 4604 4614 4636 4642 4651 4683 4697
4711 4723 4729 4739 4761 4776 4782 4792 4825 4841 4848 4858 4880 4895 4911
4916 4926 4939 4949 4982 4997 5012 5018 5027 5050 5065 5080 5086 5096 5130
5145 5159 5166 5176 5199 5214 5229 5234 5244 5267 5282 5296 5304 5314 5337
5353 5368 5377 5387 5448 5488

.IFT
511 515 523 531 535 568 587 606 624 667 685 703 721 741 767
781 801 821 841 876 894 912 930 965 984 994 1014 1034 1054 1088
1097 1101 1135 1144 1148 1182 1191 1195 1229 1238 1242 1295 1303 1305 1313
1314 1342 1350 1358 1382 1390 1393 1420 1428 1432 1460 1468 1473 1505 1513
1519 1552 1560 1567 1600 1608 1614 1643 1651 1659 1695 1703 1708 1744 1752
1761 1797 1805 1813 1834 1842 1849 1885 1893 1898 1927 1935 1938 1968 1976
1979 2006 2014 2018 2051 2083 2116 2148 2157 2165 2197 2206 2208 2240 2249
2250 2258 2265 2274 2276 2309 2318 2320 2356 2365 2371 2380 2382 2413 2422
2423 2456 2465 2467 2499 2508 2509 2541 2550 2551 2578 2605 2631 2639 2644
2652 2655 2663 2670 2678 2681 2707 2715 2716 2724 2726 2734 2738 2746 2748
2777 2805 2834 2869 2877 2879 2909 2937 2945 2947 2977 2985 2986 3015 3023

[illegible] CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

	3025	3053	3080	3107	3141	3179	3217	3249	3279	3288	3294	3327	3336	3342	[illegible]
	3359	3392	3401	3409	3418	3428	3457	3466	3471	3480	3485	3518	3554	3584	[illegible]
	3642	3650	3651	3679	3706	3714	3715	3742	3770	3778	3779	3809	3817	3818	[illegible]
	3894	3939	3947	3948	3956	3962	3970	3974	3982	3983	3991	3996	4004	4009	[illegible]
	4023	4031	4037	4081	4095	4104	4114	4123	4132	4141	4150	4159	4166	4175	4182
	4191	4201	4232	4236	4271	4280	4286	4309	4318	4324	4347	4356	4361	4385	4394
	4400	4435	4444	4451	4460	4466	4489	4498	4503	4512	4517	4526	4533	4566	4589
	4598	4604	4627	4636	4642	4674	4683	4688	4697	4702	4711	4714	4723	4729	4752
	4761	4767	4776	4782	4816	4825	4832	4841	4848	4871	4880	4886	4895	4902	491[illegible]
	4916	4939	4973	4982	4988	4997	5003	5012	5018	5041	5050	5056	5065	5071	5080
	5086	5121	5130	5136	5145	5150	5159	5166	5190	5199	5205	5214	5220	5229	5234
	5258	5267	5273	5282	5287	5296	5304	5328	5337	5344	5353	5359	5368	5377	5448
	5488														
[illegible]	424	497	502	511	520	531	540	543	559	564	573	576	578	583	592
	595	597	602	611	614	616	621	629	632	658	663	672	675	677	682
	690	693	695	700	708	711	713	718	726	729	731	736	746	749	751
	756	757	767	771	776	786	789	791	796	806	809	811	816	826	829
	831	836	846	849	867	872	881	884	886	891	899	902	904	909	917
	920	922	927	935	938	955	960	970	973	975	980	989	992	999	1002
	1004	1009	1019	1022	1024	1029	1039	1042	1044	1049	1059	1062	1078	1083	1093
	1106	1110	1125	1130	1140	1153	1157	1172	1177	1187	1200	1204	1219	1224	1234
	1247	1251	1288	1293	1300	1310	1319	1322	1334	1339	1347	1363	1367	1375	1380
	1387	1398	1402	1412	1417	1425	1437	1441	1450	1455	1465	1478	1482	1494	1499
	1510	1524	1528	1540	1545	1557	1572	1576	1588	1593	1605	1619	1623	1630	1635
	1648	1664	1668	1684	1689	1700	1713	1717	1731	1736	1749	1766	1770	1785	1790
	1802	1818	1822	1824	1829	1839	1854	1858	1875	1880	1890	1903	1907	1917	1922
	1932	1943	1947	1958	1963	1973	1984	1988	1998	2003	2004	2011	2023	2027	2038
	2043	2056	2060	2070	2075	2088	2092	2102	2107	2121	2125	2134	2139	2153	2170
	2174	2183	2188	2202	2213	2216	2226	2231	2245	2255	2270	2281	2284	2293	2298
	2314	2325	2328	2339	2344	2361	2376	2387	2390	2393	2404	2418	2428	2431	2440
	2445	2461	2472	2475	2484	2489	2504	2514	2517	2526	2531	2546	2556	2559	2567
	2572	2583	2586	2594	2599	2610	2613	2623	2628	2636	2649	2660	2675	2686	2689
	2697	2702	2712	2721	2731	2743	2753	2756	2765	2770	2782	2785	2793	2798	2810
	2813	2822	2827	2839	2842	2855	2860	2874	2884	2887	2896	2901	2914	2917	2926
	2931	2942	2952	2955	2966	2971	2982	2991	2994	3002	3007	3020	3030	3033	3043
	3048	3058	3061	3070	3075	3085	3088	3097	3102	3112	3115	3129	3134	3135	3146
	3150	3167	3172	3173	3184	3188	3205	3210	3222	3225	3237	3242	3254	3257	3267
	3272	3284	3299	3303	3314	3319	3332	3347	3364	3368	3377	3382	3397	3414	3433
	3437	3446	3451	3462	3476	3490	3494	3504	3509	3523	3527	3539	3544	3545	3559
	3563	3573	3578	3589	3593	3602	3607	3608	3619	3623	3634	3639	3647	3656	3659
	3668	3673	3683	3686	3696	3701	3711	3720	3723	3733	3738	3747	3750	3760	3765
	3775	3784	3787	3799	3804	3805	3814	3823	3827	3841	3846	3847	3856	3860	3874
	3879	3880	3889	3893	3929	3934	3935	3944	3953	3967	3979	3988	4001	4014	4028
	4042	4045	4064	4069	4070	4081	4100	4119	4137	4155	4171	4187	4206	4210	4220
	4225	4232	4241	4244	4260	4265	4276	4291	4295	4297	4302	4314	4329	4333	4335
	4340	4352	4366	4370	4372	4377	4390	4405	4409	4423	4428	4440	4456	4471	4475
	4477	4482	4494	4508	4522	4538	4542	4554	4559	4571	4575	4577	4582	4594	4609
	4613	4615	4620	4632	4647	4651	4663	4668	4679	4693	4707	4719	4734	4738	4740
	4745	4757	4772	4787	4791	4803	4808	4821	4837	4853	4857	4859	4864	4876	4891
	4907	4921	4925	4927	4932	4944	4948	4961	4966	4978	4993	5008	5023	5027	5028
	5033	5046	5061	5076	5091	5095	5108	5113	5126	5141	5155	5171	5175	5177	5182
	5195	5210	5225	5239	5243	5245	5250	5263	5278	5292	5309	5313	5315	5320	5333
	5349	5364	5382	5386	5413	5418	5419	5448	5462	5467	5468	5488			
[illegible]LIST	1	398	402	409	424	500	503	512	518	521	532	538	541	562	565
	571	574	581	584	590	593	600	603	609	612	619	622	627	630	661
	664	670	673	680	683	688	691	698	701	706	709	716	719	724	727

CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
                734   737   744   747   754   757   768   774   777   784   787   794   797   804   807
                814   817   824   827   834   837   844   847   870   873   879   882   889   892   897
                900   907   910   915   918   925   928   933   936   958   961   968   971   978   981
                987   990   997  1000  1007  1010  1017  1020  1027  1030  1037  1040  1047  1050  1057
               1060  1081  1084  1091  1095  1104  1108  1128  1131  1138  1142  1151  1155  1175  1178
               1185  1189  1198  1202  1222  1225  1232  1236  1245  1249  1291  1294  1298  1301  1308
               1311  1317  1320  1337  1340  1345  1348  1361  1365  1378  1381  1385  1388  1396  1400
               1415  1416  1423  1426  1435  1439  1453  1456  1463  1466  1476  1480  1497  1500  1508
               1511  1522  1526  1543  1546  1555  1558  1570  1574  1591  1594  1603  1606  1617  1621
               1633  1636  1646  1649  1662  1666  1687  1690  1698  1701  1711  1715  1734  1737  1747
               1750  1764  1768  1789  1791  1800  1803  1816  1820  1827  1830  1837  1840  1852  1856
               1878  1881  1888  1891  1901  1905  1920  1923  1930  1933  1941  1945  1961  1964  1971
               1974  1982  1986  2001  2004  2009  2012  2021  2025  2041  2044  2054  2058  2073  2076
               2086  2090  2105  2108  2119  2123  2137  2140  2151  2155  2168  2172  2186  2189  2200
               2204  2211  2214  2229  2232  2243  2247  2253  2256  2268  2272  2279  2282  2296  2299
               2312  2316  2323  2326  2342  2345  2359  2363  2374  2378  2385  2388  2402  2405  2416
               2420  2426  2429  2443  2446  2459  2463  2470  2473  2487  2490  2502  2506  2512  2515
               2529  2532  2544  2548  2554  2557  2570  2573  2581  2584  2597  2600  2608  2611  2626
               2629  2634  2637  2647  2650  2658  2661  2673  2676  2684  2687  2700  2703  2710  2713
               2719  2722  2729  2732  2741  2744  2751  2754  2768  2771  2780  2783  2796  2799  2808
               2811  2925  2828  2837  2840  2858  2861  2872  2875  2882  2885  2899  2902  2912  2915
               2929  2932  2940  2943  2950  2953  2969  2972  2980  2983  2989  2992  3005  3008  3018
               3021  3028  3031  3046  3049  3056  3059  3073  3076  3083  3086  3100  3103  3110  3113
               3132  3135  3144  3148  3170  3173  3182  3186  3208  3211  3220  3223  3240  3243  3252
               3255  3270  3273  3282  3286  3297  3301  3317  3320  3330  3334  3345  3349  3362  3366
               3380  3383  3395  3399  3412  3416  3431  3435  3449  3452  3460  3464  3474  3478  3488
               3492  3507  3510  3521  3525  3542  3545  3557  3561  3576  3579  3587  3591  3605  3608
               3617  3621  3637  3640  3645  3648  3654  3657  3671  3674  3681  3684  3699  3702  3709
               3712  3718  3721  3736  3739  3745  3748  3763  3766  3773  3776  3782  3785  3802  3805
               3812  3815  3821  3825  3844  3847  3854  3858  3877  3880  3887  3891  3932  3935  3942
               3945  3951  3954  3965  3968  3977  3980  3986  3989  3999  4002  4012  4015  4026  4029
               4040  4043  4067  4070  4083  4098  4102  4117  4121  4135  4139  4153  4157  4169  4173
               4185  4199  4204  4208  4223  4226  4233  4239  4242  4263  4266  4274  4278  4289  4293
               4300  4303  4312  4316  4327  4331  4338  4341  4350  4354  4364  4368  4375  4378  4388
               4392  4403  4407  4426  4429  4438  4442  4454  4458  4469  4473  4480  4483  4492  4496
               4506  4510  4520  4524  4536  4540  4557  4560  4569  4573  4580  4583  4592  4596  4607
               4611  4618  4621  4630  4634  4645  4649  4666  4669  4677  4681  4691  4695  4705  4709
               4717  4721  4732  4736  4743  4746  4755  4759  4770  4774  4785  4789  4806  4809  4819
               4823  4835  4839  4851  4655  4862  4865  4874  4878  4889  4893  4905  4909  4919  4923
               4930  4933  4942  4946  4964  4967  4976  4980  4991  4995  5006  5010  5021  5025  5031
               5034  5044  5048  5059  5063  5074  5078  5089  5093  5111  5114  5124  5128  5139  5143
               5153  5157  5169  5173  5180  5183  5193  5197  5208  5212  5223  5227  5237  5241  5248
               5251  5261  5265  5276  5280  5290  5294  5307  5311  5318  5321  5331  5335  5347  5351
               5362  5366  5380  5384  5416  5419  5449  5465  5468  5490
.MACRO             1   398  5387
.MCALL           409
.MEXIT           447
.NLIST             1   398   401   409   424   500   503   512   518   521   532   538   541   562   565
                 571   574   581   584   590   593   600   603   609   612   619   622   627   630   661
                 664   670   673   680   683   688   691   698   701   706   709   716   719   724   727
                 734   737   744   747   754   757   768   774   777   784   787   794   797   804   807
                 814   817   824   827   834   837   844   847   870   873   879   882   889   892   897
                 900   907   910   915   918   925   928   933   936   958   961   968   971   978   981
                 987   990   997  1000  1007  1010  1017  1020  1027  1030  1037  1040  1047  1050  1057
                1060  1081  1084  1091  1095  1104  1108  1128  1131  1138  1142  1151  1155  1175  1178
                1185  1189  1198  1202  1222  1225  1232  1236  1245  1249  1291  1294  1298  1301  1308
```

```
                1311    1317    1320    1337    1340    1345    1348    1361    1365    1375    1381    1385    1389    1396    1400
                1415    1418    1423    1426    1435    1439    1453    1456    1463    1466    1476    1480    1497    1500    1508
                1511    1522    1526    1543    1546    1555    1558    1570    1574    1591    1594    1603    1606    1617    1621
                1633    1636    1646    1649    1662    1666    1687    1690    1698    1701    1711    1715    1734    1737    1747
                1760    1764    1769    1788    1791    1800    1803    1816    1820    1827    1830    1837    1840    1852    1856
                1878    1881    1898    1891    1901    1905    1920    1923    1930    1933    1941    1545    1961    1964    1971
                1974    1982    1986    2001    2004    2009    2012    2021    2025    2041    2044    2054    2058    2073    2076
                2086    2090    2105    2108    2119    2123    2137    2140    2151    2155    2168    2172    2186    2189    2200
                2204    2211    2214    2229    2232    2243    2247    2253    2256    2268    2272    2279    2282    2296    2299
                2312    2316    2323    2326    2342    2345    2359    2363    2374    2378    2385    2388    2402    2405    2416
                2420    2426    2429    2443    2446    2459    2463    2470    2473    2487    2490    2502    2506    2512    2515
                2529    2532    2544    2548    2554    2557    2570    2573    2581    2584    2597    2600    2608    2611    2626
                2629    2634    2637    2647    2650    2658    2661    2673    2676    2684    2687    2700    2703    2710    2713
                2719    2722    2729    2732    2741    2744    2751    2754    2768    2771    2780    2783    2796    2799    2808
                2811    2825    2828    2837    2840    2858    2861    2872    2875    2882    2885    2899    2902    2912    2915
                2929    2932    2940    2943    2950    2953    2969    2972    2980    2983    2989    2992    3005    3008    3018
                3021    3028    3031    3046    3049    3056    3059    3073    3076    3083    3086    3100    3103    3110    3113
                3132    3135    3144    3148    3170    3173    3182    3186    3208    3211    3220    3223    3240    3243    3252
                3255    3270    3273    3282    3286    3297    3301    3317    3320    3330    3334    3345    3349    3362    3366
                3380    3383    3395    3399    3412    3416    3431    3435    3449    3452    3460    3464    3474    3478    3488
                3492    3507    3510    3521    3525    3542    3545    3557    3561    3576    3579    3587    3591    3605    3608
                3617    3621    3637    3640    3645    3648    3654    3657    3671    3674    3681    3684    3699    3702    3709
                3712    3718    3721    3736    3739    3745    3748    3763    3766    3773    3776    3782    3785    3802    3805
                3812    3815    3821    3825    3844    3847    3854    3858    3877    3880    3887    3891    3932    3935    3942
                3945    3951    3954    3965    3968    3977    3980    3986    3989    3999    4002    4012    4015    4026    4029
                4040    4043    4067    4070    4083    4098    4102    4117    4121    4135    4139    4153    4157    4169    4173
                4185    4189    4204    4208    4223    4226    4233    4239    4242    4263    4266    4274    4278    4289    4293
                4300    4303    4312    4316    4327    4331    4338    4341    4350    4354    4364    4368    4375    4378    4388
                4392    4403    4407    4426    4429    4438    4442    4454    4458    4469    4473    4480    4483    4492    4496
                4506    4510    4520    4524    4536    4540    4557    4560    4569    4573    4580    4583    4592    4596    4607
                4611    4618    4621    4630    4634    4645    4649    4666    4669    4677    4681    4691    4695    4705    4709
                4717    4721    4732    4736    4743    4746    4755    4759    4770    4774    4785    4789    4806    4809    4819
                4823    4835    4839    4851    4855    4862    4865    4874    4878    4889    4893    4905    4909    4919    4923
                4930    4933    4942    4946    4964    4967    4976    4980    4991    4995    5006    5010    5021    5025    5031
                5034    5044    5048    5059    5063    5074    5078    5089    5093    5111    5114    5124    5128    5139    5143
                5153    5157    5169    5173    5180    5183    5193    5197    5208    5212    5223    5227    5237    5241    5248
                5251    5261    5265    5276    5280    5290    5294    5307    5311    5318    5321    5331    5335    5347    5351
                5362    5366    5380    5384    5416    5419    5449    5465    5468    5490
.PADIX          5514    5532
.REM               1
.REPT            409
.SBTTL           410     421     448     500     546     562     581     600     619     635     661     680     698     716     734
                 754     774     794     814     834     853     870     889     907     925     941     958     978    1007    1027
                1047    1063    1081    1128    1175    1222    1254    1291    1337    1378    1415    1453    1497    1543    1591
                1633    1687    1734    1788    1827    1878    1920    1961    2001    2041    2073    2105    2137    2186    2229
                2296    2342    2402    2443    2487    2529    2570    2597    2626    2700    2768    2796    2825    2858    2899
                2923    2969    3005    3046    3073    3100    3132    3170    3208    3240    3270    3317    3380    3449    3507
                3542    3576    3605    3637    3671    3699    3736    3763    3802    3844    3877    3932    4067    4223    4263
                4300    4338    4375    4426    4480    4557    4580    4618    4666    4743    4806    4862    4930    4964    5031
                5111    5180    5248    5318    5416    5465
.TITLE           398
.WORD            418     426     427     428     429     430     431     432     433     437     439     439     464     465     466
                 467     468     469
```

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*,DGKAAA.SEQ/SOL/CRF PAGNUM NL:TCC DS:ERFC=S:SMAC.CC,DGKAAA.P11
RUN-TIME: 61 84 12 SECONDS
RUN-TIME RATIO: 527/158=3.3
CORE USED: 33K (65 PAGES)