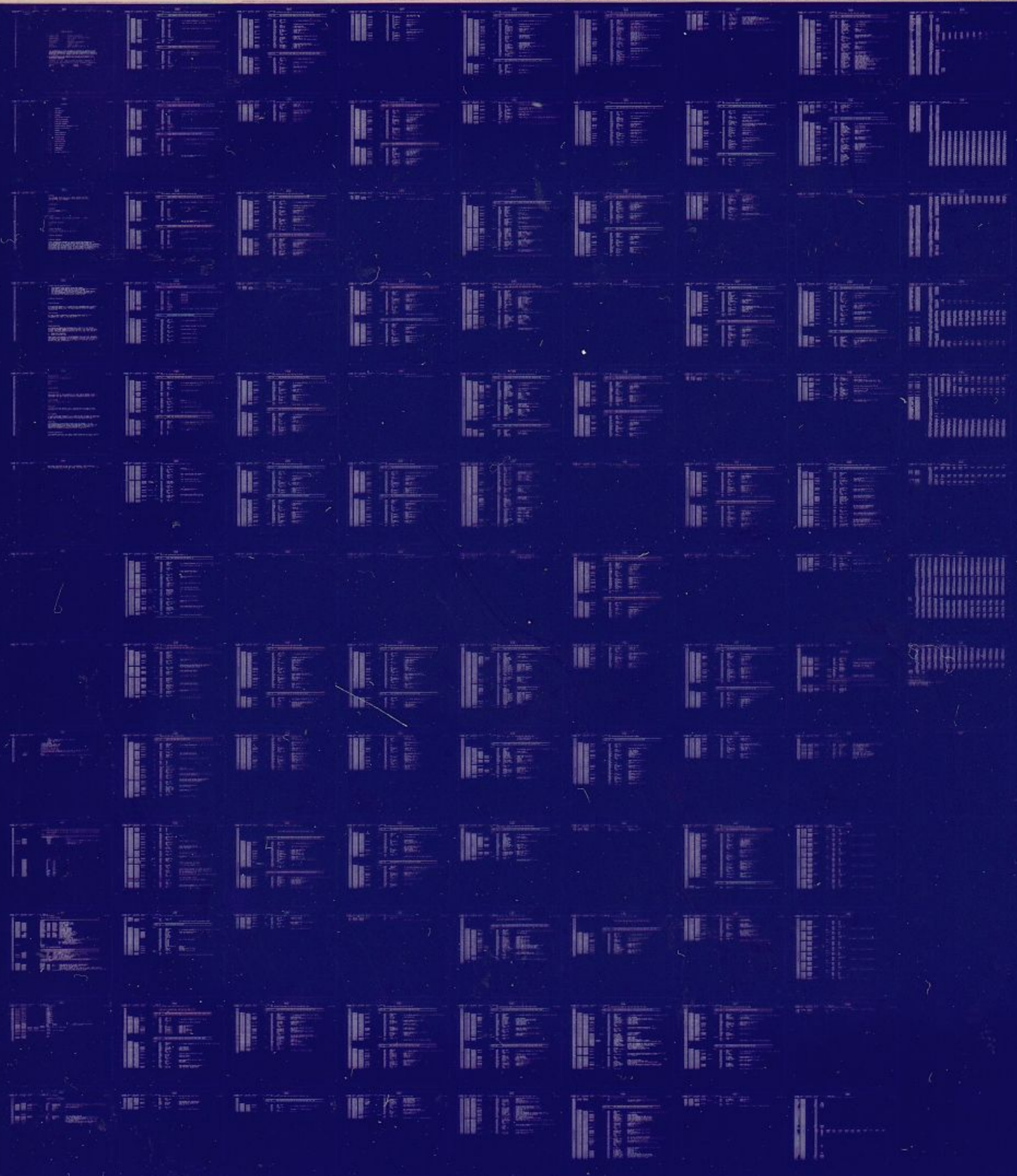
LSI 11

BASIC INSTRUCTION TESTS
MD-11-DVKAA-B

EP-DVKAA-B-DL

FICHE 1 OF 1

MAR 1978
digital
MADE IN USA



```
000000                  .REPT   0
```

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DVKAA-B-D
PRODUCT NAME: LSI-11 BASIC INSTRUCTION TESTS
DATE CREATED: FEB 1978
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: PERVEZ A. ZAKI
MODIFIED BY: BARRY SUSSMAN 01-NOV-77

CONTENTS

```
                                1.      ABSTRACT

                                        THIS PROGRAM TESTS THE LSI-11 BASIC INSTRUCTION SET
                                        IN ALL MODES. THE DIAGNOSTIC IS DESIGNED TO RUN UNDER
                                        BOTH APT. AND ACT. SYSTEMS



                                2.      REQUIREMENTS


                                2.1     EQUIPMENT

                                        LSI-11 STANDARD COMPUTER
                                        AND 4K OF MEMORY


                                2.2     STORAGE

                                        PROGRAM STORAGE - THE ROUTINES USE MEMORY 0 - 17500


                                2.3     PRELIMINARY PROGRAMS

                                        NONE


                                3.      LOADING PROCEDURE

                                        USE STANDARD PROCEDURE FOR ABS TAPES.


                                4.      STARTING PROCEDURE


                                4.1     STARTING ADDRESS

                                        AFTER LOADING THE PROGRAM IT SHOULD ALWAYS BE STARTED AT 200.
                                        IF IT IS DESIRED TO SAVE THE PASS COUNTER THEN CLEAR THE
                                        LOCATION $TESTN (I.E. LOCATION 404) AND RESTART FROM 530 OTHERWISE
                                        THE PROGRAM CAN BE RESTARTED AT 200. IF IT IS DESIRED TO GO TO A TEST
                                        OTHER THAN TEST # 0 THEN PLACE THE TEST NUMBER IN LOCATION $TESTN
                                        AND RESTART THE PROGRAM AT 530, IN WHICH CASE THE PROGRAM WILL HALT
                                        AT LOCATION 544 AND WILL WAIT FOR THE OPERATOR TO PLACE THE
                                        STARTING ADDRESS OF THE DESIRED TEST IN PC (R7) AND TYPE A P.
```

4.2 PROGRAM AND/OR OPERATOR ACTION

1) LOAD PROGRAM INTO MEMORY USING ABS LOADER.
2) TYPE 200G [THERE ARE NO SWITCH SETTINGS AND THIS DIAGNOSTIC DOES NOT USE SOFTWARE SWITCH LOCATION $SWREG]
3) THE PROGRAM WILL LOOP AND "END PASS" WILL BE TYPED AFTER THE FIRST PASS AND THEN EVERY 377 PASSES. HOWEVER TYPE OUT WILL BE SUPPRESSED IF BIT 5 OF LOCATION $ENVM IS HIGH
5) A MINIMUM OF TWO PASSES SHOULD ALWAYS BE RUN.

5. OPERATING PROCEDURE

5.1 OPERATING MODE:

AN 8 BIT BYTE $ENVM [I.E. LOCATION 421] HAS BEEN USED TO DEFINE THE OPERATING MODE. ALL TYPEOUTS CAN BE SUPPRESSED BY MAKING BIT 5 OF BYTE $ENVM HIGH, IN OTHER WORDS BY PLCING A 20000 IN LOCATION 420

5.2 TRAP CATCHER

A ".+2" - "HALT" SEQUENCE IS REPEATED FROM 0-776 TO CATCH ANY UNEXPECTED TRAPS. THUS ANY UNEXPECTED TRAPS OR INTERRUPTS WILL HALT AT THE VECTOR +2.

6. ERRORS

6.1 ERROR REPORTING

ON FINDING AN ERROR THE PROCESSOR WILL COME TO A HALT AFTER PLACING THE ERROR NUMBER IN LOCATION $FATAL [I.E. LOCATION 402]. IN MOST CASES THE COMMENTS BESIDES THE HALTS TELL WHAT WAS BEING CHECKED. IN SOME CASES THE TEST CAN GET TO A HALT VIA 2 WAYS:

1) WRONG TEST SEQUENCE
2) ERROR IN ACTUAL TEST

WHEN A HALT DOES OCCUR IT IS RECOMMENDED THAT THE TEST SEQUENCE LOCATION [I.E. LOCATION 404] BE CHECKED TO VERIFY THAT IT MATCHES THE PRESENT TEST NUMBER. IF IT DOESN'T, THEN THE HALT OCCURED BECAUSE THE TEST SEQUENCE WAS NOT CORRECT OTHERWISE THE HALT IS DUE TO AN ERROR IN THE TEST.

6.2 ERROR RECOVERY

RESTART AT 200 OR 530 (SEE SEC 4.1)

7. RESTRICTIONS

NONE

8. MISCELLANEOUS

8.1 EXECUTION TIME

EXECUTION TIME OF THE DIAGNOSTIC IS LESS THAN A SECOND, FIRST "END PASS" WILL BE TYPED OUT WITHIN A SECOND AND EVERY COSECUTIVE "END PASS" WILL BE TYPED OUT WITHIN 20 SECONDS (SEE SEC 4.2)

8.2 STACK POINTER

STACK IS INITIALLY SET TO 530

8.3 PASS COUNT

A 16 BIT LOCATION "$PASS" (I.E. LOCATION 406) IS USED TO KEEP PASS COUNT. IT CAN BE CLEARED BY RESTARTING THE PROGRAM AT 200

8.4 TEST NUMBER

A 16 BIT LOCATION "$TESTN" (I.E. LOCATION 404) IS USED TO KEEP TRACK OF THE TEST NUMBER, UPPER BYTE OF THIS LOCATION GIVES THE ITERATION NUMBER AND THE LOWER BYTE THE TEST THAT WAS BEING EXECUTED

8.5 POWER FAIL

THE DIAGNOSTIC CAN BE POWER FAILED WITH NO ERRORS. TO USE, START THE DIAGNOSTIC AS USUAL AND POWER DOWN THEN UP AT ANY TIME. THE PROGRAM SHOULD TYPE "POWER" AND RESTART AT 530 WITH TEST # 0 HOWEVER THE DIAGNOSTIC WILL NOT RECOVER IF IT IS STORED IN A MEMORY NOT CAPABLE OF HOLDING DATA WITH POWER DOWN

9. PROGRAM DESCRIPTION

THIS PROGRAM TESTS ALL THE BASIC INSTRUCTIONS OF THE LSI-11 (EXCEPT TRAP-TYPE) WHICH INCLUDES CONTROL CHIP, DATA CHIP, MICROMS, PLA,

```
                        AND OTHER CIRCUITRY ON THE LSI-11 CPU MODULE. TRAP DIAGNOSTIC
                        SHOULD ALSO BE RUN TO MAKE SURE THAT THE BASIC LSI-11 IS FUNCTIONAL
                        THIS DIAGNOSTIC DOES NOT MAKE A PASS WITH T-BIT SET.
    .ENDR
```

```
   259
```

```
   260
```

```
                                        .ABS
                                        ;; LSI-11 MACRO INSTRUCTION EXERCISER
                                        .NLIST  MC,MD,CND
                                        .LIST   ME
                                .TITLE  DVKAAB
                                ;*COPYRIGHT (C) 1975, 1977
                                ;*DIGITAL EQUIPMENT CORP.
                                ;*MAYNARD, MASS. 01754
                                ;*
                                ;*PROGRAM BY PERVEZ ZAKI
                                ;*
                                ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
                                ;*PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
                                ;*
000001                          $TN=1
160000                          $SWR=160000     ;;HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYPOUT
```

```
                        .=0

                ;;**************************************************************************
                ;       TRAP CATCHERS OF .+2 AND HALT IN LOCATIONS 0 THRU 776 (IT IS NLISTED)

                .SBTTL  ACT11 HOOKS

                ;;**************************************************************************
                ;HOOKS REQUIRED BY ACT11
001000                  $SVPC=.                 ;SAVE PC
000046                  .=46
000046  017014          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
000052                  .=52
000052  000000          .WORD   0               ;;2)SET LOC.52 TO ZERO
001000                  .=$SVPC                 ;; RESTORE PC


000400                  .=400
000000          R0      =%0
000001          R1      =%1
000002          R2      =%2
000003          R3      =%3
000004          R4      =%4
000005          R5      =%5
000006          R6      =%6
000006          SP      =%6
000007          PC      =%7
000254          CLNZ    =254
000001          ERRNM   =1
000260          NOP1    =260
000263          SEVC    =263
000273          SENVC   =273
000000          $TN     =0
000004          .TYPE   =IOT
```

```
                                        .SBTTL  APT MAILBOX-ETABLE

                                        ;;*************************************************************************
                                        .EVEN
000400                                  $MAIL:                          ;;APT MAILBOX
000400  000000                          $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
000402  000000                          $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
000404  000000                          $TESTN: .WORD   ATESTN          ;;TEST NUMBER
000406  000000                          $PASS:  .WORD   APASS           ;;PASS COUNT
000410  000000                          $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
000412  000000                          $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
000414  000000                          $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
000416  000000                          $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
000420                                  $ETABLE:                        ;;APT ENVIRONMENT TABLE
000420     000                          $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
000421     000                          $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
000422  000000                          $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
000424  000000                          $USWR:  .WORD   AUSWR           ;;USER SWITCHES
000426  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
000430                                  $ETEND:
                                        .MEXIT
                                        .SBTTL  APT PARAMETER BLOCK

                                        ;;*************************************************************************
                                        ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
                                        ;;*************************************************************************
                000430                          .$X=.   ;;SAVE CURRENT LOCATION
                000024                          .=24    ;;SET POWER FAIL TO POINT TO START OF PROGRAM
000024  000200                          200             ;;FOR APT START UP
                000044                          .=44    ;;POINT TO APT INDIRECT ADDRESS PNTR.
000044  000430                          $APTHDR ;;POINT TO APT HEADER BLOCK
                000430                          .=.$X   ;;RESET LOCATION COUNTER
                                        ;;*************************************************************************
                                        ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                                        ;INTERFACE SPEC.

000430                                  $APTHD:
000430  000000                          $HIBTS: .WORD   0               ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
000432  000400                          $MBADR: .WORD   $MAIL           ;;ADDRESS OF APT MAILBOX (BITS 0-15)
000434  000001                          $TSTM:  .WORD   1               ;;RUN TIM OF LONGEST TEST
000436  000001                          $PASTM: .WORD   1               ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
000440  000000                          $UNITM: .WORD                   ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
000442  000014                                  .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```

```
                000430                          .=$APTHD
000430          000430                  ADR:
                000432                  .=ADR+2
000432          000432                  ADR1:
                000434                  .=ADR1+2
000434          000434                  ADR2:
                000436                  .=ADR2+2
000436          000436                  DUMMY:
                000440                  .=DUMMY+2
000440          000440                  TEMP:
                000442                  .=TEMP+2
000442          000442                  TEMP1:
                000444                  .=TEMP1+2
000444          000444                  TEMP2:
                000446                  .=TEMP2+2
000446  177564                          TPS:    177564                  ; OUTPUT TTY STATUS REGISTER
000450  177566                          TPB:    177566                  ; OUTPUT BUFFER
000452  006402                  MARK2:  MARK    2
000454  005015  042440  042116  ENDPAS: .ASCIZ  <15><12>" END PASS"
000462  050040  051501  000123
                                        .EVEN
000470  005015  047520  042527  POWER:  .ASCIZ  <15><12>/POWER/
000476  000122
                                        .EVEN
000500  000023                          .BLKW   19.

                000020                  .=20
000020  017064                  TYPE
000022  000000                  0
```

NO1


```
                                        ;STARTING OF THE PROGRAM
                                        ;----------------------

                 000200                 .=200
000200  012737  017034  000024          MOV     #PWRDN,@#24     ; SERVICE POWER DOWN ROUTINE ON ANY FUTURE POWER DOWN
000206  012700  000420                  MOV     #$ETABLE,R0
000212  005040                  2$:     CLR     -(R0)           ;START CLEANING THE STACK
000214  020027  000400                  CMP     R0,#$MAIL       ;FOR INITIALIZATION
000220  101374                          BHI     2$
000222  000167  000302                  JMP     START
                                ;       ------------------------


                 000530                 .=530

000530  012706  000530          START:  MOV     #START,SP       ; SET THE STACK POINTER
000534  012705  000404                  MOV     #$TESTN,R5      ; PLACE THE ADDRESS OF LOCATION $TESTN IN R5
000540  005715                          TST     (R5)            ; CHECK THE SEQUENCE COUNTER
000542  001401                          BEQ     NOBIT           ; IF THIS IS THE STARTING OF THE TEST THEN
                                                                ; GO TO NOBIT TEST
000544  000000                          HALT                    ; OTHERWISE HALT AND WAIT FOR THE OPERATOR
                                                                ; TO START AT THE PROPER TEST NUMBER
```

```
                                ;;*******************************************************************************
                                ;*TEST: 0       CHECK BRANCH INSTRUCTIONS WITH ZERO CONDITION CODES
                                ;;*******************************************************************************

000546                          NOBIT:
000546  021527  000000                  CMP     (R5),#0
000552  001017                          BNE     CC0             ; IF IN WRONG SEQUENCE GO TO HALT AT END OF THE TEST
000554  005215                  1$:     INC     (R5)
000556  000257                          CCC                     ; ZERO CONDITION CODES, NZVC=0000
000560  103414                          BCS     CC0
000562  102413                          BVS     CC0
000564  001412                          BEQ     CC0
000566  100411                          BMI     CC0
000570  000260                          NOP1                    ; CHECK NOP1 INSTRUCTION I.E. OP-CODE 260
000572  103407                          BCS     CC0
000574  102406                          BVS     CC0
000576  001405                          BEQ     CC0
000600  100404                          BMI     CC0
000602  002403                          BLT     CC0
000604  003402                          BLE     CC0
000606  101401                          BLOS    CC0
000610  101004                          BHI     ENDCC0
000612                          CC0:
000612  012745  000001                  MOV     #1,-(R5)
000616  005245                          INC     -(R5)
000620  000000                          HALT                    ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
000622  102000                  ENDCC0: BVC     NBIT

                                ;;*******************************************************************************
                                ;*TEST: 1       CHECK BRANCH INSRUCTIONS WITH N BIT SET
                                ;;*******************************************************************************

000624                          NBIT:
000624  021527  000001                  CMP     (R5),#1
000630  001012                          BNE     CC1             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THETEST
000632  005215                  1$:     INC     (R5)
000634  000270                          SEN                     ; NBIT IS SET, NZVC=1000
000636  100007                          BPL     CC1
000640  001406                          BEQ     CC1
000642  002005                          BGE     CC1
000644  003004                          BGT     CC1
000646  103403                          BCS     CC1
000650  101402                          BLOS    CC1
000652  103401                          BLO     CC1
000654  003404                          BLE     ENDCC1
000656                          CC1:
000656  012745  000002                  MOV     #2,-(R5)
000662  005245                          INC     -(R5)
000664  000000                          HALT                    ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
000666  001000                  ENDCC1: BNE     VBIT
```

```
                                ;;*******************************************************************************
                                ;*TEST: 2        CHECK BRANCH INSRUCTIONS WITH N&V BITS SET
                                ;;*******************************************************************************

000670                          VBIT:
000670  021527  000002                  CMP     (R5),#2
000674  001014                          BNE     CC2             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
000676  005215                  1$:     INC     (R5)
000700  000270                          SEN
000702  000262                          SEV                     ; V AND N BIT SET, NZVC = 1010
000704  102010                          BVC     CC2
000706  001407                          BEQ     CC2
000710  100006                          BPL     CC2
000712  103405                          BCS     CC2
000714  002404                          BLT     CC2
000716  003403                          BLE     CC2
000720  101402                          BLOS    CC2
000722  103401                          BLO     CC2
000724  003004                          BGT     ENDCC2
000726                          CC2:
000726  012745  000003                  MOV     #3,-(R5)
000732  005245                          INC     -(R5)
000734  000000                          HALT                    ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
000736  002000                  ENDCC2: BGE     CBIT


                                ;;*******************************************************************************
                                ;*TEST: 3        CHECK BRANCH INSRUCTIONS WITH N,V&C BITS SET
                                ;;*******************************************************************************

000740                          CBIT:
000740  021527  000003                  CMP     (R5),#3
000744  001013                          BNE     CC3             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
000746  005215                  1$:     INC     (R5)
000750  000270                          SEN
000752  000262                          SEV
000754  000261                          SEC                     ; C, V, AND N BITS ARE SET, NZVC=1011
000756  001406                          BEQ     CC3
000760  100005                          BPL     CC3
000762  102004                          BVC     CC3
000764  002403                          BLT     CC3
000766  003402                          BLE     CC3
000770  101001                          BHI     CC3
000772  002004                          BGE     ZBIT
000774                          CC3:
000774  012745  000004                  MOV     #4,-(R5)
001000  005245                          INC     -(R5)
001002  000000                          HALT                    ; ONE OF THE ABOVE BRANCHES FAILED
                                                                ; OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;;*TEST: 4        CHECK BRANCH INSRUCTIONS WITH N,Z,V&C BITS SET
                                ;;*******************************************************************************

001004                          ZBIT:
001004  021527  000004                  CMP     (R5),#4
001010  001015                          BNE     CC4             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
001012  005215                          INC     (R5)
001014  000270                          SEN
001016  000262                          SEV
001020  000261                          SEC
001022  000264                          SEZ                     ; ALL BITS SET, NZVC=1111
001024  001007                          BNE     CC4
001026  100006                          BPL     CC4
001030  102005                          BVC     CC4
001032  103004                          BCC     CC4
001034  002403                          BLT     CC4
001036  003002                          BGT     CC4
001040  101001                          BHI     CC4
001042  001404                          BEQ     YESCC
001044                          CC4:
001044  012745  000005                  MOV     #5,-(R5)
001050  005245                          INC     -(R5)
001052  000000                          HALT                    ; ONE OF THE ABOVE BRANCHES FAILED
                                                                ; OR WRONG SEQUENCE


                                ;;*******************************************************************************
                                ;;*TEST: 5        CHECK BRANCH INSRUCTIONS WITH ALL THE CONDITION CODES SET
                                ;;*******************************************************************************

001054                          YESCC:
001054  021527  000005                  CMP     (R5),#5
001060  001014                          BNE     CC6             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
001062  005215                          INC     (R5)
001064  000277                          SCC                     ; NZVC=1111
001066  100011                          BPL     CC6
001070  001010                          BNE     CC6
001072  102007                          BVC     CC6
001074  103006                          BCC     CC6
001076  000240                          NOP                     ; CHECK NOP INSTRUCTION
001100  100004                          BPL     CC6
001102  001003                          BNE     CC6
001104  102002                          BVC     CC6
001106  103001                          BCC     CC6
001110  101404                          BLOS    NOTCC
001112                          CC6:
001112  012745  000006                  MOV     #6,-(R5)
001116  005245                          INC     -(R5)
001120  000000                          HALT                    ; SCC OR A BRANCH FAILED, OR WRONG SEQUENCE
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 6        CLEAR THE CONDITION CODES
                                        ;;*******************************************************************************

001122                                  NOTCC:
001122  021527  000006                          CMP     (R5),#6
001126  001013                                  BNE     CC5             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
001130  005215                                  INC     (R5)
001132  000277                                  SCC                     ; NZVC=1111
001134  000241                                  CLC                     ; NZVC=1110
001136  103407                                  BCS     CC5
001140  000242                                  CLV                     ; NZVC=1100
001142  102405                                  BVS     CC5
001144  000244                                  CLZ                     ; NZVC=1000
001146  001403                                  BEQ     CC5
001150  000250                                  CLN                     ; NZVC=0000
001152  100401                                  BMI     CC5
001154  101004                                  BHI     ENDCC5
001156                                  CC5:
001156  012745  000007                          MOV     #7,-(R5)
001162  005245                                  INC     -(R5)
001164  000000                                  HALT                    ; ONE OF THE ABOVE CLEARS FAILED OR WRONG SEQUENCE
001166  100000                          ENDCC5: BPL     BRANCH



                                        ;;*******************************************************************************
                                        ;*TEST: 7        CHECK FORWARD AND BACKWARD BRANCHES.
                                        ;;*******************************************************************************

001170                                  BRANCH:
001170  021527  000007                          CMP     (R5),#7
001174  001404                                  BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
001176  012745  000010                          MOV     #10,-(R5)
001202  005245                                  INC     -(R5)
001204  000000                                  HALT
001206  005215                          1$:     INC     (R5)
001210  000416                                  BR      4$              ; CHECK BRANCH FORWARD AND BACKWARD
001212  012745  000011                          MOV     #11,-(R5)
001216  005245                                  INC     -(R5)
001220  000000                                  HALT                    ; FORWARD BRANCH FAILED
001222  000404                          2$:     BR      3$
001224  012745  000012                          MOV     #12,-(R5)
001230  005245                                  INC     -(R5)
001232  000000                                  HALT                    ; FORWARD BRANCH FAILED
001234  000411                          3$:     BR      5$
001236  012745  000013                          MOV     #13,-(R5)
001242  005245                                  INC     -(R5)
001244  000000                                  HALT                    ; FORWARD BRANCH FAILED
001246  000765                          4$:     BR      2$
001250  012745  000014                          MOV     #14,-(R5)
001254  005245                                  INC     -(R5)
001256  000000                                  HALT                    ; BACKWARD BRANCH FAILED
001260  000400                          5$:     BR      JMP1
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 10       CHECK JMP INSTRUCTIONS FOR MODE 1
                                        ;;*******************************************************************************

001262                                  JMP1:
001262  021527  000010                          CMP     (R5),#10
001266  001033                                 BNE     ENDJP1          ; IF IN WRONG SEQUENCE GO TO HALT AT THE END OF THE TEST
001270  005215                          1$:    INC     (R5)
001272  012700  001312                         MOV     #2$,R0          ; TEST JUMP INSTRUCTION MODE 1
001276  000277                                 SCC
001300  000110                                 JMP     (R0)
001302  012745  000015                         MOV     #15,-(R5)
001306  005245                                 INC     -(R5)
001310  000000                                 HALT                    ; JUMP INSTRUCTION FAILED
001312                                  2$:
001312  100003                                 BPL     3$
001314  001002                                 BNE     3$
001316  102001                                 BVC     3$
001320  103404                                 BCS     4$
001322                                  3$:
001322  012745  000016                         MOV     #16,-(R5)
001326  005245                                 INC     -(R5)
001330  000000                                 HALT                    ; WRONG CC
001332  020027  001312                  4$:    CMP     R0,#2$
001336  001404                                 BEQ     5$              ; CONTINUE IF R0 IS OK
001340  012745  000017                         MOV     #17,-(R5)
001344  005245                                 INC     -(R5)
001346  000000                                 HALT
001350  012700  001366                  5$:    MOV     #JMP2,R0        ; TEST JUMP INSTRUCTION MODE 1
001354  000110                                 JMP     (R0)
001356                                  ENDJP1:
001356  012745  000020                         MOV     #20,-(R5)
001362  005245                                 INC     -(R5)
001364  000000                                 HALT                    ; JUMP INSTRUCTION FAILED OR WRONG SEQUENCE


                                        ;;*******************************************************************************
                                        ;*TEST: 11       CHECK JMP INSTRUCTIONS FOR MODES 2 AND 3
                                        ;;*******************************************************************************

001366                                  JMP2:
001366  021527  000011                          CMP     (R5),#11
001372  001073                                 BNE     ENDJP3          ; IF IN WRONG SEQUENCE GO TO HALT AT THE END OF TEST
001374  005215                                 INC     (R5)
001376  012700  001416                         MOV     #3$,R0          ; TEST JUMP INSTRUCTION MODE 2
001402  000277                                 SCC
001404  000120                                 JMP     (R0)+
001406  012745  000021                         MOV     #21,-(R5)
001412  005245                                 INC     -(R5)
001414  000000                                 HALT                    ; JUMP INSTRUCTION FAILED
001416                                  3$:
001416  100003                                 BPL     4$
001420  001002                                 BNE     4$
001422  102001                                 BVC     4$
001424  103404                                 BCS     5$
001426                                  4$:
```

```
 696  001426  012745  000022                    MOV     #22,-(R5)
 697  001432  005245                            INC     -(R5)
 698  001434  000000                            HALT                    ; WRONG CC
 699  001436  020027  001420          5$:       CMP     R0,#3$+2        ; IS THERE AUTO INC.?
 700  001442  001404                            BEQ     6$
 701  001444  012745  000023                    MOV     #23,-(R5)
 702  001450  005245                            INC     -(R5)
 703  001452  000000                            HALT                    ; MODE 2 FAILED FOR JMP INSTRUCTION
 704  001454  012700  001472          6$:       MOV     #JMP3,R0        ; TEST JUMP INSTRUCTION MODE 2
 705  001460  000120                            JMP     (R0)+
 706  001462  012745  000024                    MOV     #24,-(R5)
 707  001466  005245                            INC     -(R5)
 708  001470  000000                            HALT                    ; JUMP INSTRUCTION FAILED
 709
 710  001472  012767  001526  176740  JMP3:     MOV     #3$,TEMP        ; TEST JUMP INSTRUCTION MODE 3
 711  001500  012767  001546  176734            MOV     #4$,TEMP+2      ;
 712  001506  012700  000440                    MOV     #TEMP,R0
 713  001512  000277                            SCC
 714  001514  000130                            JMP     @(R0)+          ;
 715  001516  012745  000025                    MOV     #25,-(R5)
 716  001522  005245                            INC     -(R5)
 717  001524  000000                            HALT                    ; JUMP INSTRUCTION FAILED
 718  001526  027067  000000  000012  3$:       CMP     @(R0),4$        ; IS THERE AUTO INC.?
 719  001534  001404                            BEQ     4$
 720  001536  012745  000026                    MOV     #26,-(R5)
 721  001542  005245                            INC     -(R5)
 722  001544  000000                            HALT                    ; JMP INSTRUCTION FAILED IN MODE 2
 723  001546  012767  001572  176664  4$:       MOV     #JMP4,TEMP      ; TEST JUMP INSTRUCTION MODE 3
 724  001554  012700  000440                    MOV     #TEMP,R0
 725  001560  000130                            JMP     @(R0)+
 726  001562                           ENDJP3:
 727  001562  012745  000027                    MOV     #27,-(R5)
 728  001566  005245                            INC     -(R5)
 729  001570  000000                            HALT                    ; JUMP ERROR OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;*TEST: 12       TEST JUMP INSTRUCTION FOR MODE 4, 5
                                ;;*******************************************************************************

001572                          JMP4:
001572  021527  000012                  CMP     (R5),#12
001576  001075                  BNE     ENDJP5          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
001600  005215                  INC     (R5)
001602  012700  001624                  MOV     #3$,R0          ; TEST JUMP INSTRUCTION MODE 4
001606  000277                  SCC
001610  000140                  JMP     -(R0)
001612  012745  000030                  MOV     #30,-(R5)
001616  005245                  INC     -(R5)
001620  000000                  HALT                    ; JUMP INSTRUCTION FAILED
001622  000404                  BR      4$              ; JUMP SHOULD LAND HERE
001624                          3$:
001624  012745  000031                  MOV     #31,-(R5)
001630  005245                  INC     -(R5)
001632  000000                  HALT                    ; NO AUTO DECREMENT FROM JMP4
001634  022700  001622          4$:     CMP     #3$-2,R0        ; CHECK R0
001640  001404                  BEQ     5$
001642  012745  000032                  MOV     #32,-(R5)
001646  005245                  INC     -(R5)
001650  000000                  HALT
001652  012700  001672          5$:     MOV     #JMP5+2,R0      ; TEST JUMP INSTRUCTION MODE 4
001656  000140                  JMP     -(R0)
001660  012745  000033                  MOV     #33,-(R5)
001664  005245                  INC     -(R5)
001666  000000                  HALT                    ; JUMP INSTRUCTION FAILED

001670  012767  001722  176544  JMP5:   MOV     #3$,TEMP1       ; TEST JUMP INSTRUCTION MODE 5
001676  012700  000442                  MOV     #TEMP1,R0
001702  012767  001732  176530          MOV     #4$,TEMP1-2     ;
001710  000150                  JMP     @-(R0)
001712  012745  000034                  MOV     #34,-(R5)
001716  005245                  INC     -(R5)
001720  000000                  HALT                    ; JUMP INSTRUCTION FAILED
001722                          3$:
001722  012745  000035                  MOV     #35,-(R5)
001726  005245                  INC     -(R5)
001730  000000                  HALT                    ; ERROR, NO AUTO DECREMENT
001732  022700  000440          4$:     CMP     #TEMP1-2,R0     ; CHECK R0
001736  001404                  BEQ     5$
001740  012745  000036                  MOV     #36,-(R5)
001744  005245                  INC     -(R5)
001746  000000                  HALT                    ; JUMP ONSTRUCTION FAILED IN MODE 5
001750  012767  001722  176464  5$:     MOV     #3$,TEMP1       ; TEST JUMP INSTRUCTION MODE 5
001756  012700  000442                  MOV     #TEMP1,R0
001762  012767  002002  176450          MOV     #JMP6,TEMP1-2   ;
001770  000150                  JMP     @-(R0)
001772                          ENDJP5:
001772  012745  000037                  MOV     #37,-(R5)
001776  005245                  INC     -(R5)
002000  000000                  HALT                    ; JUMP ERROR OR WRONG SEQUENCE

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

```
                                        ;*TEST: 13      TEST JMP INSTRUCTION FOR MODE 6 AND 7
                                        ;;*************************************************************************

002002                                  JMP6:
002002  021527  000013                          CMP     (R5),#13
002006  001071                          BNE     ENDJP7          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
002010  005215                          INC     (R5)
002012  012703  002040                  MOV     #1$+6,R3
002016  000163  177772                  JMP     -6(R3)
002022  012745  000040                  MOV     #40,-(R5)
002026  005245                          INC     -(R5)
002030  000000                          HALT                    ; JUMP INSTRUCTION FAILED
002032  020327  002040          1$:     CMP     R3,#1$+6        ; CHECK R3
002036  001404                          BEQ     2$
002040  012745  000041                  MOV     #41,-(R5)
002044  005245                          INC     -(R5)
002046  000000                          HALT                    ; WRONG VALUE IN REGISTER AFTER JUMP MODE 6
                                                                ; OR JUMP INSTRUCTION FAILED IN MODE 6
002050  000167  000010          2$:     JMP     3$-.-4(PC)      ; TEST JUMP INSTRUCTION MODE 6
002054  012745  000042                  MOV     #42,-(R5)
002060  005245                          INC     -(R5)
002062  000000                          HALT                    ; JUMP INSTRUCTION FAILED
002064  012703  002104          3$:     MOV     #JMP7,R3        ; JUMP SHOULD LAND HERE
002070  000163  000000                  JMP     0(R3)
002074  012745  000043                  MOV     #43,-(R5)
002100  005245                          INC     -(R5)
002102  000000                          HALT                    ; JUMP INSTRUCTION FAILED

002104  012703  000440          JMP7:   MOV     #TEMP,R3
002110  012713  002130                  MOV     #1$,(R3)
002114  000173  000000                  JMP     @(R3)
002120  012745  000044                  MOV     #44,-(R5)
002124  005245                          INC     -(R5)
002126  000000                          HALT                    ; JUMP INSTRUCTION FAILED
002130  012713  002154          1$:     MOV     #3$,(R3)        ; TEST JUMP INSTRUCTION MODE 7
002134  012700  000434                  MOV     #TEMP-4,R0
002140  000170  000004                  JMP     @4(R0)
002144  012745  000045                  MOV     #45,-(R5)
002150  005245                          INC     -(R5)
002152  000000                          HALT                    ; JUMP INSTRUCTION FAILED
002154  012767  002202  176256  3$:     MOV     #JSRTST,TEMP    ; CONTINUE
002162  012700  000440                  MOV     #TEMP,R0
002166  000170  000000                  JMP     @0(R0)
002172                          ENDJP7:
002172  012745  000046                  MOV     #46,-(R5)
002176  005245                          INC     -(R5)
002200  000000                          HALT                    ; JUMP ERROR OR SEQUENCE ERROR
```

```
                                ;;*******************************************************************************
                                ;*TEST: 14       CHECK JSR AND MARK INSTRUCTIONS
                                ;;*******************************************************************************

002202                          JSRTST:
002202  021527  000014                  CMP     (R5),#14
002206  001177                         BNE     ENDJSR          ; IF IN WRONG SEQUENCE GO TO HALT AT THE END OF THE TEST
002210  005215                         INC     (R5)
002212  012706  000530                 MOV     #START,SP       ; SET UP STACK POINTER.
002216  000277                         SCC
002220  004767  000026                 JSR     PC,3$
002224                          1$:
002224  012745  000047                 MOV     #47,-(R5)
002230  005245                         INC     -(R5)
002232  000000                         HALT                    ; JSR INSTRUCTION FAILED
002234  022706  000530          2$:    CMP     #START,SP       ; HAS SP BEEN RESTORED?
002240  001441                         BEQ     JSRM
002242  012745  000050                 MOV     #50,-(R5)
002246  005245                         INC     -(R5)
002250  000000                         HALT                    ; SP WAS NOT RESTORED BY RTS INSTRUCTION
002252                          3$:
002252  100003                         BPL     4$
002254  001002                         BNE     4$
002256  102001                         BVC     4$
002260  103404                         BCS     5$
002262                          4$:
002262  012745  000051                 MOV     #51,-(R5)
002266  005245                         INC     -(R5)
002270  000000                         HALT                    ; WRONG CC
002272  022706  000526          5$:    CMP     #START-2,SP     ; WAS THE SP EFFECTED?
002276  001404                         BEQ     6$
002300  012745  000052                 MOV     #52,-(R5)
002304  005245                         INC     -(R5)
002306  000000                         HALT                    ; WRONG SP AFTER EXECUTION OF JSR INSTRUCTION
002310  022716  002224          6$:    CMP     #1$,(SP)        ; IS THE RETURN ADDRESS =1$
002314  001404                         BEQ     7$
002316  012745  000053                 MOV     #53,-(R5)
002322  005245                         INC     -(R5)
002324  000000                         HALT                    ; SP DID NOT HAVE CORRECT RETURN ADDRESS
                                                                ; AFTER EXECUTION OF JSR INSTRUCTION
002326  012716  002234          7$:    MOV     #2$,(SP)        ; SET 2$ AS THE RETURN ADDRESS
002332  000207                         RTS     PC
002334  012745  000054                 MOV     #54,-(R5)
002340  005245                         INC     -(R5)
002342  000000                         HALT                    ; RTS INSTRUCTION FAILED
002344  010546                  JSRM:  MOV     R5,-(SP)        ; MOV R5 TO STACK
002346  016746  176064                 MOV     DUMMY,-(SP)
002352  016746  176060                 MOV     DUMMY,-(SP)
002356  016746  176070                 MOV     MARK2,-(SP)     ; STORE MARK 2 ON THE STACK.
002362  010503                         MOV     R5,R3           ; SAVE R5 IN R3
002364  004467  000130                 JSR     R4,10$
002370                          1$:
002370  012745  000055                 MOV     #55,-(R5)
002374  005245                         INC     -(R5)
002376  000000                         HALT                    ; JSR INSTRUCTION FAILED
002400                          2$:
```

```
 889  002400  100003                        BPL     3$
 890  002402  001002                        BNE     3$
 891  002404  102001                        BVC     3$
 892  002406  103404                        BCS     4$
 893  002410                          3$:
 894  002410  012743  000056                MOV     #56,-(R3)
 895  002414  005243                        INC     -(R3)
 896  002416  000000                        HALT                    ; WRONG CC
 897  002420  022705  000404          4$:    CMP     #$TESTN,R5
 898  002424  001404                        BEQ     5$
 899  002426  012743  000057                MOV     #57,-(R3)
 900  002432  005243                        INC     -(R3)
 901  002434  000000                        HALT                    ; MARK INSRUCTION FAILED
 902  002436  022706  000530          5$:    CMP     #START,SP
 903  002442  001404                        BEQ     6$
 904  002444  012745  000060                MOV     #60,-(R5)
 905  002450  005245                        INC     -(R5)
 906  002452  000000                        HALT                    ; MARK INSTRUCTION FAILED
 907  002454  012701  002562          6$:    MOV     #12$,R1         ; PLACE THE ADDRESS OF 12$ IN R1
 908  002460  004011                        JSR     R0,(R1)         ; GO TO TAG 12$
 909  002462                          7$:
 910  002462  012745  000061                MOV     #61,-(R5)
 911  002466  005245                        INC     -(R5)
 912  002470  000000                        HALT                    ; JSR INSTRUCTION FAILED
 913  002472  012745  000062                MOV     #62,-(R5)
 914  002476  005245                        INC     -(R5)
 915  002500  000000                        HALT                    ; RTS BROUGHT THE PROGRAM BACK IN WRONG
 916                                                                 ; PLACE
 917  002502  022706  000530          8$:    CMP     #START,SP
 918  002506  001443                        BEQ     REGS
 919  002510  012745  000063                MOV     #63,-(R5)
 920  002514  005245                        INC     -(R5)
 921  002516  000000                        HALT                    ; STACK POINTER WAS NOT RESET
 922
 923  002520  020427  002370          10$:   CMP     R4,#1$          ; IS THE RETURN ADDRESS =1$ ?
 924  002524  001404                        BEQ     11$
 925  002526  012745  000064                MOV     #64,-(R5)
 926  002532  005245                        INC     -(R5)
 927  002534  000000                        HALT                    ; WRONG RETURN ADDRESS IN UNKAGE REGISTER R4
 928  002536  010605                  11$:   MOV     SP,R5
 929  002540  005725                        TST     (R5)+           ; SET UP ADDRESS IN R5 AT MARK 2 INSTRUCTION
 930  002542  012716  002400                MOV     #2$,(SP)        ; SET RETURN ADDRESS =2$
 931  002546  000277                        SCC
 932  002550  000205                        RTS     R5              ; RETURN USING R5 AND IN-TURN USING MARK INSTRUCTION
 933  002552  012745  000065                MOV     #65,-(R5)
 934  002556  005245                        INC     -(R5)
 935  002560  000000                        HALT                    ; RTS INSTRUCTIUN FAILED
 936
 937  002562  020027  002462          12$:   CMP     R0,#7$          ; DOES R0 CONTAIN THE RETURN ADDRESS?
 938
 939  002566  001404                        BEQ     13$
 940  002570  012745  000066                MOV     #66,-(R5)
 941  002574  005245                        INC     -(R5)
 942  002576  000000                        HALT                    ; WRONG RETURN ADDRESS IN LINKAGE REGISTER R0
 943  002600  012700  002502          13$:   MOV     #8$,R0          ; SET RETURN ADDRESS AT 8$
 944  002604  000200                        RTS     R0
```

```
945  002606                          ENDJSR:
946  002606  012745  000067                  MOV     #67,-(R5)
947  002612  005245                          INC     -(R5)
948  002614  000000                          HALT                    ; RTS INSTRUCTION FAILED OR SEQUENCE ERROR
949
950                                  ;;*****************************************************************************
951                                  ;*TEST: 15       CHECK REGISTER SELECTION
952                                  ;;*****************************************************************************
953
954  002616                          REGS:
955  002616  021527  000015                  CMP     (R5),#15
956  002622  001034                          BNE     EREGS           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
957  002624  005215                          INC     (R5)
958  002626  010667  175606                  MOV     R6,TEMP         ; SAVE THE STACK POINTER
959  002632  012700  000001                  MOV     #1,R0           ; LOAD THE REGISTERS
960  002636  012701  000004                  MOV     #4,R1           ;
961  002642  012702  000020                  MOV     #20,R2          ;
962  002646  012703  000100                  MOV     #100,R3         ;
963  002652  012704  000400                  MOV     #400,R4         ;
964  002656  005006                          CLR     R6              ;
965  002660  060006                          ADD     R0,R6           ; ADD UP THE REGISTERS
966  002662  060106                          ADD     R1,R6           ;
967  002664  060206                          ADD     R2,R6           ;
968  002666  060306                          ADD     R3,R6           ;
969  002670  060406                          ADD     R4,R6           ;
970  002672  060506                          ADD     R5,R6           ;
971  002674  022706  001131                  CMP     #$TESTN+525,R6  ; CHECK IT
972  002700  001003                          BNE     1$              ; FAILED
973  002702  016706  175532                  MOV     TEMP,R6         ; RESTORE STACK POINTER
974  002706  000406                          BR      TST80           ; CONTINUE
975  002710  016706  175524          1$:     MOV     TEMP,R6         ; RESTORE STACK POINTER
976  002714                          EREGS:
977  002714  012745  000070                  MOV     #70,-(R5)
978  002720  005245                          INC     -(R5)
979  002722  000000                          HALT                    ; REGISTER SELECTION FAILURE OR SEQUENCE ERROR
```

```
                        ;       CHECK BYTE INSTRUCTIONS, DESTINATION MODE 0 ONLY
                        ;       ----------------------------------------------

                        ;;**********************************************************************
                        ;*TEST: 16       NEW INSTRUCTIONS USED IN THIS SECTION ARE TSTB, CLRB, MOVB
                        ;;**********************************************************************

002724                  TSTB0:
002724  021527  000016          CMP     (R5),#16
002730  001404                  BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
002732  012745  000071          MOV     #71,-(R5)
002736  005245                  INC     -(R5)
002740  000000                  HALT                            ; PROGRAM IS IN WRONG SEQUENCE
002742  005215          2$:     INC     (R5)
002744  000277                  SCC
002746  105000                  CLRB    R0                      ; CLEAR THE REGISTER
002750  004737  017230          JSR     PC,@#$CC4               ; CHECK FOR CC = 4
002754  105700                  TSTB    R0                      ; CHECK IT
002756  004737  017230          JSR     PC,@#$CC4               ; CHECK FOR CC = 4
002762  112701  000377          MOVB    #377,R1                 ; LOAD THE REGISTER
002766  004737  017316          JSR     PC,@#$CC10              ; CHECK FOR CC = 10
002772  105701                  TSTB    R1                      ; CHECK IT
002774  004737  017316          JSR     PC,@#$CC10              ; CHECK FOR CC = 10


                        ;;**********************************************************************
                        ;*TEST: 17       NEW INSTRUCTIONS USED IN THIS SECTION ARE CMPB, BISB
                        ;;**********************************************************************

003000                  CMPB0:
003000  021527  000017          CMP     (R5),#17
003004  001051                  BNE     ECMPB0                  ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
003006  005215          1$:     INC     (R5)
003010  000277                  SCC
003012  152702  000377          BISB    #377,R2                 ; LOAD REGISTER
003016  004737  017336          JSR     PC,@#$CC11              ; CHECK FOR CC = 11
003022  122702  000377          CMPB    #377,R2                 ; CHECK COMPARE
003026  001404                  BEQ     2$                      ; CONTINUE IF OK
003030  012745  000072          MOV     #72,-(R5)
003034  005245                  INC     -(R5)
003036  000000                  HALT                            ; BISB OR CMPB INSTRUCTION FAILED
003040  112700  000077          MOVB    #77,R0
003044  120002                  CMPB    R0,R2                   ; CHECK IT AGAIN
003046  100004                  BPL     3$                      ; CONTINUE IF OK
003050  012745  000073          MOV     #73,-(R5)
003054  005245                  INC     -(R5)
003056  000000                  HALT                            ; CMPB INSTRUCTION FAILED (WRONG CC)
003060  120200          3$:     CMPB    R2,R0                   ; ONCE MORE
003062  100404                  BMI     4$                      ; CONTINUE IF OK
003064  012745  000074          MOV     #74,-(R5)
003070  005245                  INC     -(R5)
003072  000000                  HALT                            ; CMPB INSTRUCTION FAILED (WRONG CC)
003074  112702  000377          MOVB    #377,R2         4$:     ; LOAD REGISTER, SIGN EXTEND
003100  122702  000377          CMPB    #377,R2                 ; CHECK IF BYTE INSTRUCTION
```

```
 1036  003104  001404                        BEQ     5$                ; CONTINUE IF OK
 1037  003106  012745  000075                MOV     #75,-(R5)
 1038  003112  005245                        INC     -(R5)
 1039  003114  000000                        HALT                      ; CMPB BECAME CMP INSTRUCTION
 1040  003116  112702  000377        5$:     MOVB    #377,R2           ; LOAD REGISTER, SIGN EXTEND
 1041  003122  120227  000377                CMPB    R2,#377           ; CHECK IF BYTE INSTRUCTION
 1042  003126  001404                        BEQ     BIC60             ; CONTINUE IF OK
 1043  003130                        ECMPB0:
 1044  003130  012745  000076                MOV     #76,-(R5)
 1045  003134  005245                        INC     -(R5)
 1046  003136  000000                        HALT                      ; WRONG CC OR WRONG SEQUENCE
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 20      NEW INSTRUCTIONS USED IN THIS SECTION ARE BICB, BITB
                                        ;;*******************************************************************************

003140                                  BICB0:
003140  021527  000020                          CMP     (R5),#20
003144  001404                          BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
003146  012745  000077                  MOV     #77,-(R5)
003152  005245                          INC     -(R5)
003154  000000                          HALT                            ; PROGRAM IS IN WRONG SEQUENCE
003156  005215                  2$:     INC     (R5)
003160  112703  000377                  MOVB    #377,R3                 ; LOAD REGISTER
003164  112700  000252                  MOVB    #252,R0                 ; PLACE #252 IN R0
003170  000277                          SCC
003172  140003                          BICB    R0,R3                   ; CLEAR EVERY OTHER BIT
003174  004737  017146                  JSR     PC,@#$CC1               ; CHECK FOR CC = 1
003200  130003                          BITB    R0,R3                   ; CHECK IT
003202  001404                          BEQ     4$                      ; CONTINUE IF OK
003204  012745  000100                  MOV     #100,-(R5)
003210  005245                          INC     -(R5)
003212  000000                          HALT                            ; BICB OR BITB INSTRUCTION FAILED
003214  132703  000125          4$:     BITB    #125,R3                 ; CHECK IT
003220  004737  017146                  JSR     PC,@#$CC1               ; CHECK FOR CC = 1
003224  150003                          BISB    R0,R3                   ; SET THE BITS THAT WERE CLEARED
003226  100404                          BMI     6$
003230  012745  000101                  MOV     #101,-(R5)
003234  005245                          INC     -(R5)
003236  000000                          HALT                            ; BISB INSTRUCTION FAILED
003240  142703  000177          6$:     BICB    #177,R3                 ; CLEAR ALL THE BITS EXCEPT FOR SIGN
003244  004737  017336                  JSR     PC,@#$CC11              ; CHECK FOR CC = 11
003250  132703  000377                  BITB    #377,R3                 ; CHECK IT
003254  004737  017336                  JSR     PC,@#$CC11              ; CHECK FOR CC = 11


                                        ;;*******************************************************************************
                                        ;*TEST: 21      NEW INSTRUCTIONS USED IN THIS SECTION ARE INCB, DECB
                                        ;;*******************************************************************************

003260                                  INCB0:
003260  021527  000021                          CMP     (R5),#21
003264  001404                          BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
003266  012745  000102                  MOV     #102,-(R5)
003272  005245                          INC     -(R5)
003274  000000                          HALT                            ; PROGRAM IS IN WRONG SEQUENCE
003276  005215                  1$:     INC     (R5)
003300  112704  000177                  MOVB    #177,R4                 ; R4 = 177
003304  000261                          SEC
003306  105204                          INCB    R4                      ; ADD ONES INTO REG. 4
003310  004737  017400                  JSR     PC,@#$CC13              ; CHECK FOR CC = 13
003314  112704  000376                  MOVB    #376,R4
003320  105204                          INCB    R4
003322  004737  017336                  JSR     PC,@#$CC11              ; CHECK FOR CC = 11
003326  105204                          INCB    R4
003330  004737  017252                  JSR     PC,@#$CC5               ; CHECK FOR CC = 5
003334  105204                          INCB    R4
```

```
                        T21     NEW INSTRUCTIONS USED IN THIS SECTION ARE INCB, DECB                    SEQ 0028

1103  003336  004737  017146            JSR     PC,@#$CC1       ; CHECK FOR CC = 1
1104  003342  122704  000001            CMPB    #1,R4           ; CHECK IT
1105  003346  001404                    BEQ     2$              ; CONTINUE IF OK
1106  003350  012745  000103            MOV     #103,-(R5)
1107  003354  005245                    INC     -(R5)
1108  003356  000000                    HALT                    ; INCB INSTRUCTION FAILED
1109  003360  000261            2$:     SEC
1110  003362  105304                    DECB    R4              ; SUBTRACT ONES FROM REG. 4
1111  003364  004737  017252            JSR     PC,@#$CC5       ; CHECK FOR CC = 5
1112  003370  105304                    DECB    R4
1113  003372  004737  017336            JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1114  003376  012704  000200            MOV     #200,R4
1115  003402  105304                    DECB    R4
1116  003404  004737  017206            JSR     PC,@#$CC3       ; CHECK FOR CC = 3
1117  003410  105304                    DECB    R4
1118  003412  004737  017146            JSR     PC,@#$CC1       ; CHECK FOR CC = 1
```

```
                                    ;;*******************************************************************************
                                    ;*TEST: 22      NEW INSTRUCTION IN THIS SECTION IS COMB
                                    ;;*******************************************************************************

003416                              COMBO:
003416  021527  000022                      CMP     (R5),#22
003422  001404                              BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
003424  012745  000104                      MOV     #104,-(R5)
003430  005245                              INC     -(R5)
003432  000000                              HALT                    ; PROGRAM IS IN WRONG SEQUENCE
003434  005215                      1$:     INC     (R5)
003436  112703  000252                      MOVB    #252,R3         ; LOAD EVERY OTHER BIT
003442  000277                              SCC
003444  105103                              COMB    R3              ; 1'S COMPLEMENT
003446  004737  017146                      JSR     PC,@#$CC1       ; CHECK FOR CC = 1
003452  122703  000125                      CMPB    #125,R3         ; CHECK IT
003456  001404                              BEQ     2$              ; CONTINUE IF OK
003460  012745  000105                      MOV     #105,-(R5)
003464  005245                              INC     -(R5)
003466  000000                              HALT                    ; COMB INSTRUCTION FAILED
003470  000277                      2$:     SCC
003472  105103                              COMB    R3              ; COMPLEMENT BACK
003474  004737  017336                      JSR     PC,@#$CC11      ; CHECK FOR CC = 11
003500  122703  000252                      CMPB    #252,R3         ; CHECK IT
003504  001404                              BEQ     3$              ; CONTINUE IF OK
003506  012745  000106                      MOV     #106,-(R5)
003512  005245                              INC     -(R5)
003514  000000                              HALT                    ; COMB INSTRUCTION FAILED
003516  012703  000377              3$:     MOV     #377,R3
003522  000277                              SCC
003524  105103                              COMB    R3
003526  004737  017252                      JSR     PC,@#$CC5       ; CHECK FOR CC = 5


                                    ;;*******************************************************************************
                                    ;*TEST: 23      NEW INSTRUCTION IN THIS SECTION IS NEGB
                                    ;;*******************************************************************************

003532                              NEGB0:
003532  021527  000023                      CMP     (R5),#23
003536  001025                              BNE     ENEGB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
003540  005215                      1$:     INC     (R5)
003542  112700  000001                      MOVB    #1,R0           ; LOAD THE REGISTER
003546  105400                              NEGB    R0              ; 2'S COMPLEMENT
003550  004737  017336                      JSR     PC,@#$CC11      ; CHECK FOR CC = 11
003554  122700  000377                      CMPB    #377,R0         ; CHECK IT
003560  001404                              BEQ     2$              ; CONTINUE IF OK
003562  012745  000107                      MOV     #107,-(R5)
003566  005245                              INC     -(R5)
003570  000000                              HALT                    ; NEGB INSTRUCTION FAILED
003572  012700  000200              2$:     MOV     #200,R0
003576  105400                              NEGB    R0              ; 2'S COMPLEMENT
003600  004737  017400                      JSR     PC,@#$CC13      ; CHECK FOR CC = 13
003604  122700  000200                      CMPB    #200,R0         ; CHECK IT
003610  001404                              BEQ     ROLB0           ; CONTINUE IF OK
```

E03


```
 1175  003612                          ENEGB0:
 1176  003612  012745  000110                  MOV     #110,-(R5)
 1177  003616  005245                          INC     -(R5)
 1178  003620  000000                          HALT                    ; WRONG RESULT IN R0 OR WRONG SEQUENCE
```

```
1179                                        ;;*****************************************************************************
1180                                        ;*TEST: 24      NEW INSTRUCTION IN THIS SECTION IS ROLB
1181                                        ;;*****************************************************************************
1182
1183  003622                                ROLB0:
1184  003622  021527  000024                       CMP     (P5),#24
1185  003626  001026                               BNE     EROLB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
1186  003630  005215                               INC     (R5)
1187  003632  112701  000040                       MOVB    #40,R1          ; LOAD REGISTER
1188  003636  000257                               CCC                     ; CLEAR FLAGS
1189  003640  106101                               ROLB    R1              ; SHIFT
1190  003642  106101                               ROLB    R1              ;
1191  003644  004737  017360                       JSR     PC,@#$CC12      ; CHECK FOR CC = 12
1192  003650  122701  000200                       CMPB    #200,R1         ; CHECK IT
1193  003654  001404                               BEQ     1$              ; CONTINUE IF OK
1194  003656  012745  000111                       MOV     #111,-(R5)
1195  003662  005245                               INC     -(R5)
1196  003664  000000                               HALT                    ; ROLB INSTRUCTION FAILED
1197  003666  106101                       1$:     ROLB    R1              ; SHIFT
1198  003670  004737  017274                       JSR     PC,@#$CC7       ; CHECK FOR CC = 7
1199  003674  106101                               ROLB    R1              ; SHIFT
1200  003676  122701  000001                       CMPB    #1,R1           ; CHECK IT
1201  003702  001404                               BEQ     RORB0           ; CONTINUE IF OK
1202  003704                                EROLB0:
1203  003704  012745  000112                       MOV     #112,-(R5)
1204  003710  005245                               INC     -(R5)
1205  003712  000000                               HALT                    ; WRONG RESULT IN R1 OR WRONG SEQUENCE
1206
1207
1208                                        ;;*****************************************************************************
1209                                        ;*TEST: 25      NEW INSTRUCTION IN THIS SECTION IS RORB
1210                                        ;;*****************************************************************************
1211
1212  003714                                RORB0:
1213  003714  021527  000025                       CMP     (R5),#25
1214  003720  001026                               BNE     ERORB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
1215  003722  005215                               INC     (R5)
1216  003724  112702  000004                       MOVB    #4,R2           ; LOAD REGISTER
1217  003730  000257                               CCC                     ; CLEAR FLAGS
1218  003732  106002                               RORB    R2              ; SHIFT
1219  003734  106002                               RORB    R2              ;
1220  003736  122702  000001                       CMPB    #1,R2           ; CHECK IT
1221  003742  001404                               BEQ     1$              ; CONTINUE IF OK
1222  003744  012745  000113                       MOV     #113,-(R5)
1223  003750  005245                               INC     -(R5)
1224  003752  000000                               HALT                    ; RORB INSTRUCTION FAILED
1225  003754  106002                       1$:     RORB    R2              ; SHIFT
1226  003756  004737  017274                       JSR     PC,@#$CC7       ; CHECK FOR CC = 7
1227  003762  106002                               RORB    R2              ; SHIFT
1228  003764  004737  017360                       JSR     PC,@#$CC12      ; CHECK FOR CC = 12
1229  003770  122702  000200                       CMPB    #200,R2         ; CHECK IT
1230  003774  001404                               BEQ     ASLB0           ; CONTINUE IF OK
1231  003776                                ERORB0:
1232  003776  012745  000114                       MOV     #114,-(R5)
1233  004002  005245                               INC     -(R5)
1234  004004  000000                               HALT
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 26      NEW INSTRUCTION IN THIS SECTION IS ASLB
                                        ;;*******************************************************************************

004006                                  ASLB0:
004006  021527  000026                          CMP     (R5),#26
004012  001404                          BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
004014  012745  000115                  MOV     #115,-(R5)
004020  005245                          INC     -(R5)
004022  000000                          HALT                            ; PROGRAM IS IN WRONG SEQUENCE
004024  005215                  2$:     INC     (R5)
004026  112703  000040                  MOVB    #40,R3                  ; LOAD REGISTER
004032  000257                          CCC                             ; CLEAR FLAGS
004034  106303                          ASLB    R3                      ; SHIFT
004036  106303                          ASLB    R3                      ;
004040  004737  017360                  JSR     PC,@#$CC12              ; CHECK FOR CC = 12
004044  122703  000200                  CMPB    #200,R3                 ; CHECK IT
004050  001404                          BEQ     4$                      ; CONTINUE IF OK
004052  012745  000116                  MOV     #116,-(R5)
004056  005245                          INC     -(R5)
004060  000000                          HALT                            ; ASLB INSTRUCTION FAILED
004062  106303                  4$:     ASLB    R3                      ; SHIFT
004064  004737  017274                  JSR     PC,@#$CC7               ; CHECK FOR CC = 7
004070  106303                          ASLB    R3                      ; SHIFT
004072  004737  017230                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
                                        ;;*******************************************************************************
                                        ;*TEST: 27      NEW INSTRUCTION IN THIS SECTION IS ASRB
                                        ;;*******************************************************************************

004076                                  ASRB0:
004076  021527  000027                          CMP     (R5),#27
004102  001034                          BNE     EASRB0                  ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
004104  005215                  1$:     INC     (R5)
004106  112704  000004                  MOVB    #4,R4                   ; LOAD REGISTER
004112  000257                          CCC                             ; CLEAR FLAGS
004114  106204                          ASRB    R4                      ; SHIFT
004116  106204                          ASRB    R4                      ;
004120  122704  000001                  CMPB    #1,R4                   ; CHECK IT
004124  001404                          BEQ     2$                      ; CONTINUE IF OK
004126  012745  000117                  MOV     #117,-(R5)
004132  005245                          INC     -(R5)
004134  000000                          HALT                            ; ASRB INSTRUCTION FAILED
004136  106204                  2$:     ASRB    R4                      ; SHIFT
004140  004737  017274                  JSR     PC,@#$CC7               ; CHECK FOR CC = 7
004144  106204                          ASRB    R4                      ; SHIFT
004146  004737  017230                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
004152  112703  000202                  MOVB    #202,R3                 ; LOAD REGISTER
004156  106203                          ASRB    R3                      ; SHIFT
004160  106203                          ASRB    R3                      ;
004162  004737  017336                  JSR     PC,@#$CC11              ; CHECK FOR CC = 11
004166  122703  000340                  CMPB    #340,R3                 ; CHECK IT
004172  001404                          BEQ     ADCB0                   ; CONTINUE IF OK
004174                                  EASRB0:
004174  012745  000120                  MOV     #120,-(R5)
004200  005245                          INC     -(R5)
```

```
  1291  004202  000000                        HALT
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 30      NEW INSTRUCTION IN THIS SECTION IS ADCB
                                        ;;*******************************************************************************

004204                                  ADCB0:
004204  021527  000030                          CMP     (P5),#30
004210  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
004212  012745  000121                  MOV     #121,-(R5)
004216  005245                          INC     -(R5)
004220  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
004222  005215                  2$:     INC     (R5)
004224  105000                          CLRB    R0              ; CLEAR THE REGISTER
004226  000257                          CCC                     ; CLEAR FLAGS
004230  105500                          ADCB    R0              ; ADD C BIT = 0
004232  004737  017230                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
004236  000261                          SEC                     ; C=1
004240  105500                          ADCB    R0              ; ADD C BIT=1
004242  000261                          SEC                     ; C=1
004244  105500                          ADCB    R0              ; AGAIN
004246  004737  017126                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
004252  122700  000002                  CMPB    #2,R0           ; CHECK IT
004256  001404                          BEQ     4$              ; CONTINUE IF OK
004260  012745  000122                  MOV     #122,-(R5)
004264  005245                          INC     -(R5)
004266  000000                          HALT                    ; ADCB INSTRUCTION FAILED
004270  112700  000177          4$:     MOVB    #177,R0         ; LOAD LARGEST POSITVE NUMBER
004274  000261                          SEC                     ; C=1
004276  105500                          ADCB    R0              ; ADD C BIT=1
004300  004737  017360                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
004304  122700  000200                  CMPB    #200,R0         ; CHECK IT
004310  001404                          BEQ     6$              ; CONTINUE IF OK
004312  012745  000123                  MOV     #123,-(R5)
004316  005245                          INC     -(R5)
004320  000000                          HALT                    ; ADCB INSTRUCTION FAILED
004322  112700  000377          6$:     MOVB    #377,R0         ; LOAD -1
004326  000261                          SEC                     ; C=1
004330  105500                          ADCB    R0              ; ADD C BIT=1
004332  004737  017252                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5


                                        ;;*******************************************************************************
                                        ;*TEST: 31      NEW INSTRUCTION IN THIS SECTION IS SBCB
                                        ;;*******************************************************************************

004336                                  SBCB0:
004336  021527  000031                          CMP     (R5),#31
004342  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
004344  012745  000124                  MOV     #124,-(R5)
004350  005245                          INC     -(R5)
004352  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
004354  005215                  1$:     INC     (R5)            ; TEST IS IN WRONG SEQUENCE
004356  112701  000003                  MOVB    #3,R1           ; LOAD REGISTER
004362  000257                          CCC                     ; CLEAR FLAGS
004364  105601                          SBCB    R1              ; SUBTRACT C BIT=0
004366  004737  017126                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
```

T31 NEW INSTRUCTION IN THIS SECTION IS SBCB SEQ 0035

```
1348  004372  122701  000003          CMPB    #3,R1           ; CHECK IT
1349  004376  001404                  BEQ     2$              ; CONTINUE IF OK
1350  004400  012745  000125          MOV     #125,-(P5)
1351  004404  005245                  INC     -(R5)
1352  004406  000000                  HALT                    ; SBCB INSTRUCTION FAILED
1353  004410  000261          2$:     SEC                     ; C=1
1354  004412  105601                  SBCB    R1              ; SUBTRACT C BIT=1
1355  004414  000261                  SEC                     ; C=1
1356  004416  105601                  SBCB    R1              ;
1357  004420  004737  017126          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
1358  004424  122701  000001          CMPB    #1,R1           ; CHECK IT
1359  004430  001404                  BEQ     3$              ; CONTINUE IF OK
1360  004432  012745  000126          MOV     #126,-(R5)
1361  004436  005245                  INC     -(R5)
1362  004440  000000                  HALT                    ; SBCB INSTRUCTION FAILED
1363  004442  000261          3$:     SEC                     ; C=1
1364  004444  105601                  SBCB    R1              ; SUBTRACT C BIT=1
1365  004446  004737  017230          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
1366  004452  000261                  SEC                     ; C=1
1367  004454  105601                  SBCB    R1              ; SUBTRACT C BIT = 1
1368  004456  004737  017336          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1369  004462  122701  000377          CMPB    #377,R1         ; CHECK IT
1370  004466  001404                  BEQ     4$              ; CONTINUE IF OK
1371  004470  012745  000127          MOV     #127,-(R5)
1372  004474  005245                  INC     -(R5)
1373  004476  000000                  HALT                    ; SBCB INSTRUCTION FAILED
1374  004500  112701  000200          MOVB    #200,R1         ; LOAD R1
1375  004504  000261          4$:     SEC                     ; C=1
1376  004506  105601                  SBCB    R1              ; SUBTRACT C BIT = 1
1377  004510  004737  017166          JSR     PC,@#$CC2       ; CHECK FOR CC = 2
```

```
                                        ;
                                        ;               CHECK WORD INSTRUCTIONS, DESTINATION MODE 0 ONLY
                                        ;               ----------------------------------------------
                                        ;;**************************************************************************
                                        ;*TEST: 32      NEW INSTRUCTIONS USED IN THIS SECTION ARE TST, CLR, MOV
                                        ;;**************************************************************************
004514                                  TST0:
004514  021527  000032                          CMP     (R5),#32
004520  001404                          BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
004522  012745  000130                  MOV     #130,-(R5)
004526  005245                          INC     -(R5)
004530  000000                          HALT                            ; TEST IS IN WRONG SEQUENCE
004532  005215                  1$:     INC     (R5)
004534  000277                          SCC
004536  005000                          CLR     R0                      ; CLEAR THE REGISTER
004540  004737  017230                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
004544  005700                          TST     R0                      ; CHECK IT
004546  004737  017230                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
004552  012704  177777                  MOV     #177777,R4              ; LOAD THE REGISTER
004556  010401                          MOV     R4,R1
004560  004737  017316                  JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004564  005701                          TST     R1                      ; CHECK IT
004566  004737  017316                  JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004572  020401                          CMP     R4,R1                   ; CHECK R1 TO CONTAIN PROPER DATA
004574  001404                          BEQ     2$
004576  012745  000131                  MOV     #131,-(R5)
004602  005245                          INC     -(R5)
004604  000000                          HALT
004606  000263                  2$:     SEVC                            ; SET V & C BITS
004610  010000                          MOV     R0,R0
004612  004767  012434                  JSR     PC,$CC5
                                        ;;**************************************************************************
                                        ;*TEST: 33      NEW INSTRUCTIONS USED IN THIS SECTION ARE CMP, BIS
                                        ;;**************************************************************************
004616                                  CMP0:
004616  021527  000033                          CMP     (R5),#33
004622  001026                          BNE     ECMP0                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
004624  005215                  1$:     INC     (R5)
004626  012700  177777                  MOV     #177777,R0              ; LOAD REGISTER
004632  050002                          BIS     R0,R2                   ; CHECK THE BIS INSTRUCTION
004634  004737  017316                  JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004640  020002                          CMP     R0,R2                   ; CHECK COMPARE
004642  001404                          BEQ     2$                      ; CONTINUE IF OK
004644  012745  000132                  MOV     #132,-(R5)
004650  005245                          INC     -(R5)
004652  000000                          HALT                            ; BIS OR CMP INSTRUCTION FAILED
```

```
1434  004654  022702  000077          2$:     CMP     #77,R2          ; CHECK IT AGAIN
1435  004660  100004                          BPL     3$              ; CONTINUE IF OK
1436  004662  012745  000133                  MOV     #133,-(P5)
1437  004666  005245                          INC     -(R5)
1438  004670  000000                          HALT                    ; CMP INSTRUCTION FAILED [WRONG CC]
1439  004672  020227  000077          3$:     CMP     R2,#77          ; ONCE MORE
1440  004676  100404                          BMI     BICC            ; CONTINUE IF OK
1441  004700                          ECMPO:
1442  004700  012745  000134                  MOV     #134,-(R5)
1443  004704  005245                          INC     -(R5)
1444  004706  000000                          HALT                    ; WRONG CC OR WRONG SEQUENCE
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 34      NEW INSTRUCTIONS USED IN THIS SECTION ARE BIC, BIT
                                        ;;*******************************************************************************

004710                                  BICO:
004710  021527  000034                          CMP     (R5),#34
004714  001053                                  BNE     EBICO           ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
004716  005215                                  INC     (R5)
004720  012703  177777                        MOV     #177777,R3      ; LOAD REGISTER
004724  012700  000440                        MOV     #TEMP,R0        ; PLACE THE ADDRESS OF LOCATION TEMP IN R0
004730  012710  125252                        MOV     #125252,(R0)    ; SET (R0)
004734  000277                              SCC
004736  041003                              BIC     (R0),R3         ; CLEAR EVERY OTHER BIT
004740  004737  017146                      JSR     PC,@#$CC1       ; CHECK FOR CC = 1
004744  031003                              BIT     (R0),R3         ; CHECK IT
004746  001404                              BEQ     1$              ; CONTINUE IF OK
004750  012745  000135                      MOV     #135,-(R5)
004754  005245                              INC     -(R5)
004756  000000                              HALT                    ; BIC OR BIT INSTRUCTION FAILED
004760  032703  052525          1$:         BIT     #52525,R3       ; CHECK IT
004764  004737  017146                      JSR     PC,@#$CC1       ; CHECK FOR CC = 1
004770  052703  125252                      BIS     #125252,R3      ; SET THE BITS THAT WERE CLEARED
004774  100404                              BMI     2$              ; CONTINUE IF OK
004776  012745  000136                      MOV     #136,-(R5)
005002  005245                              INC     -(R5)
005004  000000                              HALT                    ; BIT OR BIS INSTRUCTION FAILED
005006  042703  077777          2$:         BIC     #77777,R3       ; CLEAR ALL THE BITS EXCEPT FOR SIGN
005012  004737  017336                      JSR     PC,@#$CC11      ; CHECK FOR CC = 11
005016  012700  177777                      MOV     #177777,R0
005022  030003                              BIT     R0,R3           ; CHECK IT
005024  004737  017336                      JSR     PC,@#$CC11      ; CHECK FOR CC = 11
005030  000263                              SEVC                    ; SET V & C BITS
005032  040000                              BIC     R0,R0
005034  004737  017252                      JSR     PC,@#$CC5       ; CHECK CC = 5
005040  005700                              TST     R0              ; CHECK R0 TO CONTAIN 0
005042  001404                              BEQ     INCO
005044                                  EBICO:
005044  012745  000137                      MOV     #137,-(R5)
005050  005245                              INC     -(R5)
005052  000000                              HALT                    ; BIC FAILED OR SEQUENCE ERROR
```

```
 1487
 1488                                   ;;*****************************************************************************
 1489                                   ;*TEST: 35      NEW INSTRUCTIONS USED IN THIS SECTION ARE INC, DEC
 1490                                   ;;*****************************************************************************
 1491
 1492  005054                           INCO:
 1493  005054  021527  000035                  CMP     (R5),.#35
 1494  005060  001404                          BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
 1495  005062  012745  000140                  MOV     #140,-(R5)
 1496  005066  005245                          INC     -(R5)
 1497  005070  000000                          HALT                            ; PROGRAM IS IN WRONG SEQUENCE
 1498  005072  005215                  2$:     INC     (R5)
 1499  005074  012704  077777                  MOV     #77777,R4               ; R4=77777
```

```
 1500  005100  000261                        SEC
 1501  005102  005204                        INC     R4                      ; ADD ONES INTO REG. 4
 1502  005104  004737  017400                JSR     PC,@#$CC13              ; CHECK FOR CC = 13
 1503  005110  012704  177776                MOV     #177776,R4
 1504  005114  005204                        INC     R4
 1505  005116  004737  017336                JSR     PC,@#$CC11              ; CHECK FOR CC = 11
 1506  005122  005204                        INC     R4
 1507  005124  004737  017252                JSR     PC,@#$CC5               ; CHECK FOR CC = 5
 1508  005130  005204                        INC     R4
 1509  005132  004737  017146                JSR     PC,@#$CC1               ; CHECK FOR CC = 1
 1510  005136  022704  000001                CMP     #1,R4                   ; CHECK IT
 1511  005142  001404                        BEQ     4$                      ; FAILED
 1512  005144  012745  000141                MOV     #141,-(R5)
 1513  005150  005245                        INC     -(R5)
 1514  005152  000000                        HALT                            ; INC INSTRUCTION FAILED
 1515  005154  000261              4$:       SEC
 1516  005156  005304                        DEC     R4                      ; SUBTRACT ONES FROM REG. 4
 1517  005160  004737  017252                JSR     PC,@#$CC5               ; CHECK FOR CC = 5
 1518  005164  005304                        DEC     R4
 1519  005166  004737  017336                JSR     PC,@#$CC11              ; CHECK FOR CC = 11
 1520  005172  012704  100000                MOV     #100000,R4
 1521  005176  005304                        DEC     R4                      ;
 1522  005200  004737  017206                JSR     PC,@#$CC3               ; CHECK FOR CC = 3
 1523  005204  005304                        DEC     R4                      ;
 1524  005206  004737  017146                JSR     PC,@#$CC1               ; CHECK FOR CC = 1
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 36      NEW INSTRUCTION IN THIS SECTION IS COM
                                        ;;*******************************************************************************

005212                                  COM0:
005212  021527  000036                          CMP     (R5),#36
005216  001404                          1$:     BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
005220  012745  000142                          MOV     #142,-(R5)
005224  005245                                  INC     -(R5)
005226  000000                                  HALT                    ; TEST IS IN WRONG SEQUENCE
005230  005215                                  INC     (R5)
005232  012703  125252                          MOV     #125252,R3      ; LOAD EVERY OTHER BIT
005236  000277                                  SCC
005240  005103                                  COM     R3              ; 1'S COMPLEMENT
005242  004737  017146                          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
005246  022703  052525                          CMP     #52525,R3       ; CHECK IT
005252  001404                                  BEQ     2$              ; CONTINUE IF OK
005254  012745  000143                          MOV     #143,-(R5)
005260  005245                                  INC     -(R5)
005262  000000                                  HALT                    ; COM INSTRUCTION FAILED
005264  000277                          2$:     SCC
005266  005103                                  COM     R3              ; COMPLEMENT BACK
005270  004737  017336                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
005274  022703  125252                          CMP     #125252,R3      ; CHECK IT
005300  001404                                  BEQ     3$              ; CONTINUE IF OK
005302  012745  000144                          MOV     #144,-(R5)
005306  005245                                  INC     -(R5)
005310  000000                                  HALT                    ; COM INSTRUCTION FAILED
005312  012703  177777                  3$:     MOV     #177777,R3
005316  000277                                  SCC
005320  005103                                  COM     R3
005322  004737  017252                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
                                        ;;*******************************************************************************
                                        ;*TEST: 37      NEW INSTRUCTION IN THIS SECTION IS NEG
                                        ;;*******************************************************************************

005326                                  NEG0:
005326  021527  000037                          CMP     (R5),#37
005332  001025                                  BNE     ENEG0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
005334  005215                          1$:     INC     (R5)
005336  012700  000001                          MOV     #1,R0           ; LOAD THE REGISTER
005342  005400                                  NEG     R0              ; 2'S COMPLEMENT
005344  004737  017336                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
005350  022700  177777                          CMP     #177777,R0      ; CHECK IT
005354  001404                                  BEQ     2$              ; CONTINUE IF OK
005356  012745  000145                          MOV     #145,-(R5)
005362  005245                                  INC     -(R5)
005364  000000                                  HALT                    ; NEG INSTRUCTION FAILED
005366  012700  100000                  2$:     MOV     #100000,R0
005372  005400                                  NEG     R0              ; 2'S COMPLEMENT
005374  004737  017400                          JSR     PC,@#$CC13      ; CHECK FOR CC = 13
005400  022700  100000                          CMP     #100000,R0      ; CHECK IT
005404  001404                                  BEQ     ROL0            ; CONTINUE IF OK
```

T37 NEW INSTRUCTION IN THIS SECTION IS NEG SEQ 0042

```
1581  005406                          ENEGO:
1582  005406  012745  000146                MOV     #146,-(R5)
1583  005412  005245                        INC     -(R5)
1584  005414  000000                        HALT                    ; WRONG RESULT IN R0 OR WRONG SEQUENCE
```

```
1585
1586                                    ;;*******************************************************************************
1587                                    ;*TEST: 40      NEW INSTRUCTION IN THIS SECTION IS ROL
1588                                    ;;*******************************************************************************
1589   005416                           ROLO:
1590   005416  021527  000040                  CMP     (R5),#40
1591   005422  001026                          BNE     EROLO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
1592   005424  005215                          INC     (R5)
1593   005426  012701  020000                  MOV     #20000,R1       ; LOAD REGISTER
1594   005432  000257                          CCC                     ; CLEAR FLAGS
1595   005434  006101                          ROL     R1              ; SHIFT
1596   005436  006101                          ROL     R1              ;
1597   005440  004737  017360                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
1598   005444  022701  100000                  CMP     #100000,R1      ; CHECK IT
1599   005450  001404                          BEQ     1$              ; CONTINUE IF OK
1600   005452  012745  000147                  MOV     #147,-(R5)
1601   005456  005245                          INC     -(R5)
1602   005460  000000                          HALT                    ; ROL INSTRUCTION FAILED
1603   005462  006101                  1$:     ROL     R1              ; SHIFT
1604   005464  004737  017274                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
1605   005470  006101                          ROL     R1              ; SHIFT
1606   005472  022701  000001                  CMP     #1,R1           ; CHECK IT
1607   005476  001404                          BEQ     RORO            ; CONTINUE IF OK
1608   005500                           EROLO:
1609   005500  012745  000150                  MOV     #150,-(R5)
1610   005504  005245                          INC     -(R5)
1611   005506  000000                          HALT                    ; WRONG RESULT IN R1 OR WRONG SEQUENCE
1612
1613
1614
1615                                    ;;*******************************************************************************
1616                                    ;*TEST: 41      NEW INSTRUCTION IN THIS SECTION IS ROR
1617                                    ;;*******************************************************************************
1618
1619   005510                           RORO:
1620   005510  021527  000041                  CMP     (R5),#41
1621   005514  001026                          BNE     ERORO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
1622   005516  005215                          INC     (R5)
1623   005520  012702  000004                  MOV     #4,R2           ; LOAD REGISTER
1624   005524  000257                          CCC                     ; CLEAR FLAGS
1625   005526  006002                          ROR     R2              ; SHIFT
1626   005530  006002                          ROR     R2              ;
1627   005532  022702  000001                  CMP     #1,R2           ; CHECK IT
1628   005536  001404                          BEQ     1$              ; CONTINUE IF OK
1629   005540  012745  000151                  MOV     #151,-(R5)
1630   005544  005245                          INC     -(R5)
1631   005546  000000                          HALT                    ; ROR INSTRUCTION FAILED
1632   005550  006002                  1$:     ROR     R2              ; SHIFT
1633   005552  004737  017274                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
1634   005556  006002                          ROR     R2              ; SHIFT
1635   005560  004737  017360                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
1636   005564  022702  100000                  CMP     #100000,R2      ; CHECK IT
1637   005570  001404                          BEQ     ASLO            ; CONTINUE IF OK
1638   005572                           ERORO:
1639   005572  012745  000152                  MOV     #152,-(R5)
1640   005576  005245                          INC     -(R5)
```

F04

NEW INSTRUCTION IN THIS SECTION IS ROR SEQ 0044

```
  1641  005600  000000                    HALT                    ; WRONG RESULT IN R2 OR WRONG SEQUENCE
```

```
1642                                    ;;*************************************************************************
1643                                    ;*TEST: 42      NEW INSTRUCTION IN THIS SECTION IS ASL
1644                                    ;;*************************************************************************
1645
1646  005602                            ASLO:
1647  005602  021527  000042                   CMP     (R5),#42
1648  005606  001404                           BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
1649  005610  012745  000153                   MOV     #153,-(R5)
1650  005614  005245                           INC     -(R5)
1651  005616  000000                           HALT                    ; PROGRAM IS IN WRONG SEQUENCE
1652  005620  005215                    2$:    INC     (R5)
1653  005622  012703  020000                   MOV     #20000,R3       ; LOAD REGISTER
1654  005626  000257                           CCC                     ; CLEAR FLAGS
1655  005630  006303                           ASL     R3              ; SHIFT
1656  005632  006303                           ASL     R3              ;
1657  005634  004737  017360                   JSR     PC,@#$CC12      ; CHECK FOR CC = 12
1658  005640  022703  100000                   CMP     #100000,R3      ; CHECK IT
1659  005644  001404                           BEQ     4$              ; CONTINUE IF OK
1660  005646  012745  000154                   MOV     #154,-(R5)
1661  005652  005245                           INC     -(R5)
1662  005654  000000                           HALT                    ; ASL INSTRUCTION FAILED
1663  005656  006303                    4$:    ASL     R3              ; SHIFT
1664  005660  004737  017274                   JSR     PC,@#$CC7       ; CHECK FOR CC = 7
1665  005664  006303                           ASL     R3              ; SHIFT
1666  005666  004737  017230                   JSR     PC,@#$CC4       ; CHECK FOR CC = 4
1667
1668                                    ;;*************************************************************************
1669                                    ;*TEST: 43      NEW INSTRUCTION IN THIS SECTION IS ASR
1670                                    ;;*************************************************************************
1671
1672  005672                            ASRO:
1673  005672  021527  000043                   CMP     (R5),#43
1674  005676  001034                           BNE     EASRO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
1675  005700  005215                    1$:    INC     (R5)
1676  005702  012704  000004                   MOV     #4,R4           ; LOAD REGISTER
1677  005706  000257                           CCC                     ; CLEAR FLAGS
1678  005710  006204                           ASR     R4              ; SHIFT
1679  005712  006204                           ASR     R4              ;
1680  005714  022704  000001                   CMP     #1,R4           ; CHECK IT
1681  005720  001404                           BEQ     2$              ; CONTINUE IF OK
1682  005722  012745  000155                   MOV     #155,-(R5)
1683  005726  005245                           INC     -(R5)
1684  005730  000000                           HALT                    ; ASR INSTRUCTION FAILED
1685  005732  006204                    2$:    ASR     R4              ; SHIFT
1686  005734  004737  017274                   JSR     PC,@#$CC7       ; CHECK FOR CC = 7
1687  005740  006204                           ASR     R4              ; SHIFT
1688  005742  004737  017230                   JSR     PC,@#$CC4       ; CHECK FOR CC = 4
1689  005746  012703  100002                   MOV     #100002,R3      ; LOAD REGISTER
1690  005752  006203                           ASR     R3              ; SHIFT
1691  005754  006203                           ASR     R3              ;
1692  005756  004737  017336                   JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1693  005762  022703  160000                   CMP     #160000,R3      ; CHECK IT
1694  005766  001404                           BEQ     ADCO            ; CONTINUE IF OK
1695  005770                            EASRO:
1696  005770  012745  000156                   MOV     #156,-(R5)
1697  005774  005245                           INC     -(R5)
```

H04


```
  1698  005776  000000                        HALT                    ; WRONG RESULT IN R3 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;*TEST: 44      NEW INSTRUCTION IN THIS SECTION IS ADC
                                ;;*******************************************************************************

006000                          ADC0:
006000  021527  000044                  CMP     (P5),#44
006004  001404                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
006006  012745  000157          MOV     #157,-(R5)
006012  005245                  INC     -(R5)
006014  000000                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
006016  005215          2$:     INC     (R5)
006020  005000                  CLR     R0              ; CLEAR THE REGISTER
006022  000257                  CCC                     ; CLEAR FLAGS
006024  005500                  ADC     R0              ; ADD C BIT = 0
006026  004737  017230          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
006032  000261                  SEC                     ; C=1
006034  005500                  ADC     R0              ; ADD C BIT=1
006036  000261                  SEC                     ; C=1
006040  005500                  ADC     R0              ; AGAIN
006042  004737  017126          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
006046  022700  000002          CMP     #2,R0           ; CHECK IT
006052  001404                  BEQ     4$              ; CONTINUE IF OK
006054  012745  000160          MOV     #160,-(R5)
006060  005245                  INC     -(R5)
006062  000000                  HALT                    ; ADC INSTRUCTION FAILED
006064  012700  077777          MOV     #77777,R0       ; LOAD LARGEST POSITVE NUMBER
006070  000261                  SEC                     ; C=1
006072  005500                  ADC     R0              ; ADD C BIT=1
006074  004737  017360          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
006100  022700  100000          CMP     #100000,R0      ; CHECK IT
006104  001404                  BEQ     6$              ; FAILED
006106  012745  00016!          MOV     #161,-(R5)
006112  005245                  INC     -(R5)
006114  000000                  HALT                    ; ADC INSTRUCTION FAILED
006116  012700  177777          MOV     #-1,R0          ; LOAD -1
006122  000261                  SEC                     ; C=1
006124  005500                  ADC     R0              ; ADD C BIT=1
006126  004737  017252          JSR     PC,@#$CC5       ; CHECK FOR CC = 5


                                ;;*******************************************************************************
                                ;*TEST: 45      NEW INSTRUCTION IN THIS SECTION IS SBC
                                ;;*******************************************************************************

006132                          SBC0:
006132  021527  000045                  CMP     (R5),#45
006136  001404                  BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
006140  012745  000162          MOV     #162,-(R5)
006144  005245                  INC     -(R5)
006146  000000                  HALT                    ; TEST IS IN WRONG SEQUENCE
006150  005215          1$:     INC     (R5)
006152  012701  000003          MOV     #3,R1           ; LOAD REGISTER
006156  000257                  CCC                     ; CLEAR FLAGS
006160  005601                  SBC     R1              ; SUBTRACT C BIT=0
006162  004737  017126          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
```

```
1755  006166  022701  000003                CMP     #3,R1           ; CHECK IT
1756  006172  001404                       BEQ     2$              ; CONTINUE IF OK
1757  006174  012745  000163                MOV     #163,-(P5)
1758  006200  005245                       INC     -(R5)
1759  006202  000000                       HALT                    ; SBC INSTRUCTION FAILED
1760  006204  000261              2$:      SEC                     ; C=1
1761  006206  005601                       SBC     R1              ; SUBTRACT C BIT=1
1762  006210  000261                       SEC                     ; C=1
1763  006212  005601                       SBC     R1              ;
1764  006214  004737  017126               JSR     PC,@#$CC0       ; CHECK FOR CC = 0
1765  006220  022701  000001               CMP     #1,R1           ; CHECK IT
1766  006224  001404                       BEQ     3$              ; CONTINUE IF OK
1767  006226  012745  000164                MOV     #164,-(R5)
1768  006232  005245                       INC     -(R5)
1769  006234  000000                       HALT                    ; SBC INSTRUCTION FAILED
1770  006236  000261              3$:      SEC                     ; C=1
1771  006240  005601                       SBC     R1              ; SUBTRACT C BIT=1
1772  006242  004737  017230               JSR     PC,@#$CC4       ; CHECK FOR CC = 4
1773  006246  000261                       SEC                     ; C=1
1774  006250  005601                       SBC     R1              ; SUBTRACT C BIT = 1
1775  006252  004737  017336               JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1776  006256  022701  177777               CMP     #-1,R1          ; CHECK IT
1777  006262  001404                       BEQ     4$              ; CONTINUE IF F OK
1778  006264  012745  000165                MOV     #165,-(R5)
1779  006270  005245                       INC     -(R5)
1780  006272  000000                       HALT                    ; SBC INSTRUCTION FAILED
1781  006274  012701  100000              4$:      MOV     #100000,R1      ; LOAD R1
1782  006300  000261                       SEC                     ; C=1
1783  006302  005601                       SBC     R1              ; SUBTRACT C BIT = 1
1784  006304  004737  017166               JSR     PC,@#$CC2       ; CHECK FOR CC = 2
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 46      NEW INSTRUCTION IN THIS SECTION IS SXT
                                        ;;*******************************************************************************

                                        SXTO:
006310 021527 000046                            CMP     (P5),#46
006314 001024                                   BNE     ESXTO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
006316 005215                           1$:     INC     (R5)
006320 005002                                   CLR     R2              ; CLEAR REGISTER
006322 000277                                   SCC
006324 000254                                   CLNZ
006326 006702                                   SXT     R2              ; SIGN EXTEND
006330 004737 017252                            JSR     PC,@#$CC5       ; CHECK FOR CC = 5
006334 005702                                   TST     R2              ; REG. 2 SHOULD STILL BE 0
006336 001404                                   BEQ     2$              ; CONTINUE IF OK
006340 012745 000166                            MOV     #166,-(R5)
006344 005245                                   INC     -(R5)
006346 000000                                   HALT                    ; SXT INSTRUCTION FAILED
006350 000273                           2$:     SENVC                   ; SET N, V & C BITS
006352 006702                                   SXT     R2              ; SIGN EXTEND
006354 004737 017336                            JSR     PC,@#$CC11      ; CHECK FOR CC = 11
006360 022702 177777                            CMP     #-1,R2          ; REG. 2 SHOULD NOW HAVE -1
006364 001404                                   BEQ     SWABO           ; CONTINUE IF OK
006366                                  ESXTO:
006366 012745 000167                            MOV     #167,-(R5)
006372 005245                                   INC     -(R5)
006374 000000                                   HALT                    ; WRONG RESULT IN R2 OR WRONG SEQUENCE


                                        ;;*******************************************************************************
                                        ;*TEST: 47      NEW INSTRUCTION IN THIS SECTION IS SWAB
                                        ;;*******************************************************************************

006376                                  SWABO:
006376 021527 000047                            CMP     (R5),#47
006402 001031                                   BNE     ESWABO          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
006404 005215                                   INC     (R5)
006406 012703 125125                            MOV     #125125,R3      ; LOAD BIT PATTERN INTO REGISTER
006412 000277                                   SCC
006414 000250                                   CLN
006416 000303                                   SWAB    R3              ; SWAP BYTES OF REGISTER
006420 004737 017316                            JSR     PC,@#$CC10      ; CHECK FOR CC = 10
006424 022703 052652                            CMP     #52652,R3       ; CHECK IT
006430 001404                                   BEQ     1$              ; CONTINUE IF OK
006432 012745 000170                            MOV     #170,-(R5)
006436 005245                                   INC     -(R5)
006440 000000                                   HALT                    ; SWAB INSTRUCTION FAILED
006442 012703 000377                    1$:     MOV     #377,R3
006446 000277                                   SCC
006450 000244                                   CLZ
006452 000303                                   SWAB    R3
006454 004737 017230                            JSR     PC,@#$CC4       ; CHECK FOR CC = 4
006460 022703 177400                            CMP     #177400,R3
006464 001404                                   BEQ     XORO
006466                                  ESWABO:
```

NEW INSTRUCTION IN THIS SECTION IS SWAB SEQ 0050

```
  1841  006466  012745  000171            MOV     #171,-(R5)
  1842  006472  005245                    INC     -(R5)
  1843  006474  000000                    HALT                    ; WRONG RESULT IN R3 OR WRONG SEQUENCE
```

```
1844                                            ;;*****************************************************************************
1845                                            ;*TEST: 50      NEW INSTRUCTION IN THIS SECTION IS XOR
1846                                            ;;*****************************************************************************
1847
1848  006476                                    XORO:
1849  006476  021527  000050                           CMP     (PS),#50
1850  006502  001034                                   BNE     EXORO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
1851  006504  005215                                   INC     (R5)
1852  006506  012704  177777                           MOV     #-1,R4          ; LOAD REGISTERS
1853  006512  012703  177777                           MOV     #-1,R3          ;
1854  006516  000277                                   SCC
1855  006520  074403                                   XOR     R4,R3           ; SHOULD PRODUCE 0'S IN REG. 3
1856  006522  004737  017252                           JSR     PC,@#$CC5       ; CHECK FOR CC = 5
1857  006526  012703  077777                           MOV     #77777,R3
1858  006532  010400                                   MOV     R4,R0           ; PLACE A -1 IN R0
1859  006534  000263                                   SEVC                    ; SET V & C BITS
1860  006536  000244                                   CLZ
1861  006540  074003                                   XOR     R0,R3
1862  006542  004737  017336                           JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1863  006546  012702  125252                           MOV     #125252,R2      ; LOAD REGISTERS
1864  006552  012704  052525                           MOV     #52525,R4       ;
1865  006556  000277                                   SCC
1866  006560  074203                                   XOR     R2,R4           ; SHOULD PRODUCE ALL 1'S IN REG. 4
1867  006562  004737  017336                           JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1868  006566  022704  177777                           CMP     #-1,R4          ; CHECK IT
1869  006572  001404                                   BEQ     ADDO            ; CONTINUE IF OK
1870  006574                                    EXORO:
1871  006574  012745  000172                           MOV     #172,-(R5)
1872  006600  005245                                   INC     -(R5)
1873  006602  000000                                   HALT                    ; WRONG RESULT IN R4 OR WRONG SEQUENCE
1874
1875
1876
1877                                            ;;*****************************************************************************
1878                                            ;*TEST: 51      NEW INSTRUCTION IN THIS SECTION IS ADD
1879                                            ;;*****************************************************************************
1880
1881  006604                                    ADDO:
1882  006604  021527  000051                           CMP     (R5),#51
1883  006610  001055                                   BNE     EADDO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
1884  006612  005215                                   INC     (R5)
1885  006614  012701  021421                           MOV     #21421,R1       ; LOAD REGISTERS
1886  006620  060101                                   ADD     R1,R1           ; ADD
1887  006622  004737  017126                           JSR     PC,@#$CC0       ; CHECK FOR CC = 0
1888  006626  022701  043042                           CMP     #43042,R1       ; CHECK IT
1889  006632  001404                                   BEQ     1$              ; CONTINUE IF OK
1890  006634  012745  000173                           MOV     #173,-(R5)
1891  006640  005245                                   INC     -(R5)
1892  006642  000000                                   HALT                    ; ADD INSTRUCTION FAILED
1893  006644  012700  156357                   1$:     MOV     #-21421,R0      ; LOAD REGISTERS
1894  006650  060000                                   ADD     R0,R0           ; ADD
1895  006652  004737  017336                           JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1896  006656  022700  134736                           CMP     #-43042,R0      ; CHECK IT
1897  006662  001404                                   BEQ     2$              ; CONTINUE IF OK
1898  006664  012745  000174                           MOV     #174,-(R5)
1899  006670  005245                                   INC     -(R5)
```

```
                        T51     NEW INSTRUCTION IN THIS SECTION IS ADD

 1900  006672  000000                HALT                    ; ADD INSTRUCTION FAILED
 1901  006674  012702  100000    2$: MOV     #100000,R2      ; LOAD REGISTERS
 1902  006700  060202              ADD     R2,R2           ; ADD SHOULD RESULT AS 0'S
 1903  006702  004737  017274      JSR     PC,@#$CC7       ;CHECK FOR CC = 7
 1904  006706  012704  021421      MOV     #21421,R4       ; LOAD REGISTERS
 1905  006712  012701  156357      MOV     #-21421,R1      ;
 1906  006716  060401              ADD     R4,R1           ; ADD SHOULD RESULT AS 0'S
 1907  006720  001404              BEQ     3$              ; CONTINUE IF OK
 1908  006722  012745  000175      MOV     #175,-(R5)
 1909  006726  005245              INC     -(R5)
 1910  006730  000000              HALT                    ; ADD INSTRUCTION FAILED
 1911  006732  005404    3$:       NEG     R4              ; SWITCH SOURCE AND DESTINATION
 1912  006734  012701  021421      MOV     #21421,R1       ;
 1913  006740  060104              ADD     R1,R4           ; SHOULD RESULT AS 0'S
 1914  006742  001404              BEQ     SUBO            ; CONTINUE IF OK
 1915  006744                EADDO:
 1916  006744  012745  000176      MOV     #176,-(R5)
 1917  006750  005245              INC     -(R5)
 1918  006752  000000              HALT                    ; WRONG RESULT IN R1 OR WRONG SEQUENCE
```

```
                                        ;;*****************************************************************************
                                        ;*TEST: 52      NEW INSTRUCTION IN THIS SECTION IS SUB
                                        ;;*****************************************************************************

006754                                  SUB0:
006754  021527  000052                          CMP     (R5),#52
006760  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
006762  012745  000177                  MOV     #177,-(R5)
006766  005245                          INC     -(R5)
006770  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
006772  005215                  2$:     INC     (R5)
006774  012702  021421                  MOV     #21421,R2       ; LOAD REGISTERS
007000  012703  156357                  MOV     #-21421,R3      ;
007004  160203                          SUB     R2,R3           ; RESULT SHOULD=-43042
007006  004737  017316                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
007012  022703  134736                  CMP     #-43042,R3      ; CHECK IT
007016  001404                          BEQ     4$              ; CONTINUE IF OK
007020  012745  000200                  MOV     #200,-(R5)
007024  005245                          INC     -(R5)
007026  000000                          HALT                    ; SUB INSTRUCTION FAILED
007030  012703  021421          4$:     MOV     #21421,R3       ; LOAD REGISTER
007034  010204                          MOV     R2,R4           ; NOW R4 = #21421
007036  160403                          SUB     R4,R3           ; RESULT SHOULD=0
007040  001404                          BEQ     6$
007042  012745  000201                  MOV     #201,-(R5)
007046  005245                          INC     -(R5)
007050  000000                          HALT                    ; SUB NSTRUCTION FAILED
007052  012703  177777          6$:     MOV     #-1,R3          ; LOAD REGISTERS
007056  012702  077777                  MOV     #77777,R2       ; LOAD REGISTERS
007062  160302                          SUB     R3,R2           ; RESULT SHOULD BE 100000 AND OVERFLOW
007064  004737  017400                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
007070  022702  100000                  CMP     #100000,R2      ; CHECK IT
007074  001404                          BEQ     8$              ; CONTINUE IF OK
007076  012745  000202                  MOV     #202,-(R5)
007102  005245                          INC     -(R5)
007104  000000                          HALT                    ; SUB INSTRUCTION FAILED
007106  012704  177777          8$:     MOV     #-1,R4
007112  160304                          SUB     R3,R4
007114  004737  017230                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4


                                        ;;*****************************************************************************
                                        ;*TEST: 53      NEW INSTRUCTIONS IN THARE SECTION IS MTPS & MFPS
                                        ;;*****************************************************************************

007120                                  PSW:
007120  021527  000053                          CMP     (R5),#53
007124  001032                          BNE     EPSW            ; IF IN WRONG SEQUENCE THEN GO TO HLT AT THE END OF THE
007126  005215                  1$:     INC     (R5)
007130  012701  177777                  MOV     #177777,R1
007134  005000                          CLR     R0
007136                                  MTPS    R0              ; SET PSW TO 0
007136  106400                          .WORD   106400!..C
007140  004737  017126                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
007144                                  MFPS    R1              ; MOVE PSW TO R1
007144  106701                          .WORD   106700!..C
```

```
                        T53     NEW INSTRUCTIONS IN THARE SECTION IS MTPS & MFPS                                SEQ 0054

 1975  007146  001404                        BEQ     2$                      ; CONTINUE IF BIT 8 OF PSW WAS EXTENDED IN R1
 1976  007150  012745  000203                MOV     #203,-(R5)
 1977  007154  005245                        INC     -(R5)
 1978  007156  000000                        HALT                            ; MTPS OR MFPS INSTRUCTION FAILED
 1979  007160  004737  017230        2$:     JSR     PC,@#$CC4               ; CHECK FOR CC = 4
 1980  007164  012700  000377                MOV     #377,R0
 1981  007170                                MTPS    R0                      ; SET PSW TO 357 SINCE MTPS DOES NOT SET T BIT
 1982  007170  106400                        .WORD   106400!..C
 1983  007172  004737  017420                JSR     PC,@#$CC17              ; CHECK FOR CC = 17
 1984  007176                                MFPS    R1                      ; MOVE PSW TO R1
 1985  007176  106701                        .WORD   106700!..C
 1986  007200  004737  017336                JSR     PC,@#$CC11              ; CHECK FOR CC = 11 (C BIT SHOULD NOT BE EFFECTED BY MFP
 1987  007204  022701  177757                CMP     #177757,R1              ; CHECK TO SEE IF BIT 8 OF PSW WAS EXTENDED THRU R1
 1988  007210  001404                        BEQ     MODE0
 1989  007212                        EPSW:
 1990  007212  012745  000204                MOV     #204,-(R5)
 1991  007216  005245                        INC     -(R5)
 1992  007220  000000                        HALT                            ; MTPS OR MFPS INSTRUCTION FAILED OR WRONG SEQUENCE
```

```
                                ;               LSI-11 INSTRUCTIONS NOT MODE 0
                                ;               ------------------------


                                ;;*****************************************************************************
                                ;*TEST: 54       CHECK MODES 0 & 1 USING THE MOVB AND MOV INSTRUCTIONS
                                ;,*****************************************************************************

007222                          MODE0:
007222  021527  000054                  CMP     (R5),#54
007226  001063                          BNE     EMODE0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
007230  005215                          INC     (R5)
007232  112700  000252                  MOVB    #252,R0         ; LOAD REGISTERS
007236  110001                          MOVB    R0,R1           ;
007240  110102                          MOVB    R1,R2           ;
007242  122702  000252                  CMPB    #252,R2         ; CHECK IT
007246  001404                          BEQ     1$              ; OK, CONTINUE
007250  012745  000205                  MOV     #205,-(R5)
007254  005245                          INC     -(R5)
007256  000000                          HALT                    ; MOV INSTRUCTION FAILED IN MODE 0
007260  012700  125252          1$:     MOV     #125252,R0      ; LOAD REGISTERS
007264  010001                          MOV     R0,R1           ;
007266  010102                          MOV     R1,R2           ;
007270  022702  125252                  CMP     #125252,R2      ; CHECK IT
007274  001404                          BEQ     MODE1           ; OK, CONTINUE
007276  012745  000206                  MOV     #206,-(R5)
007302  005245                          INC     -(R5)
007304  000000                          HALT                    ; MOV INSTRUCTION FAILED IN MODE 0

007306  012700  000440          MODE1:  MOV     #TEMP,R0        ; LOAD ADDRESSES INTO REGS.
007312  012701  000442                  MOV     #TEMP1,R1       ;
007316  012702  000444                  MOV     #TEMP2,R2       ;
007322  005067  171116                  CLR     TEMP2           ; START CLEAN
007326  112710  000125                  MOVB    #125,(R0)       ; LOAD THE LOCATIONS
007332  111011                          MOVB    (R0),(R1)       ; TEMP ---> TEMP1
007334  111112                          MOVB    (R1),(R2)       ; TEMP1 ---> TEMP2
007336  122767  000125  171100          CMPB    #125,TEMP2      ; CHECK IT
007344  001404                          BEQ     1$              ; OK, CONTINUE
007346  012745  000207                  MOV     #207,-(R5)
007352  005245                          INC     -(R5)
007354  000000                          HALT                    ; MOV INSTRUCTION FAILED IN MODE 1
007356  012710  052525          1$:     MOV     #52525,(R0)     ; LOAD THE LOCATIONS
007362  011011                          MOV     (R0),(R1)       ; TEMP ---> TEMP1
007364  011112                          MOV     (R1),(R2)       ; TEMP1 ---> TEMP2
007366  022767  052525  171050          CMP     #52525,TEMP2    ; CHECK IT
007374  001404                          BEQ     MODE2           ; OK, CONTINUE
007376                          EMODE0:
007376  012745  000210                  MOV     #210,-(R5)
007402  005245                          INC     -(R5)
007404  000000                          HALT                    ; MOV INSTRUCTION FAILED IN MODE 1
                                                                ; OR WPONG SEQUENCE


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

```
2049                                    ;*TEST: 55      CHECK MODE 2 USING THE MOVB AND MOV INSTRUCTIONS
2050                                    ;;**************************************************************************
2051
2052  007406                            MODE2:
2053  007406  021527  000055                   CMP     (R5),#55
2054  007412  001050                           BNE     EMODE2          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
2055  007414  005215                           INC     (R5)
2056  007416  012700  000440                   MOV     #TEMP,R0        ; LOAD ADDRESSES
2057  007422  012701  000442                   MOV     #TEMP1,R1       ;
2058  007426  012702  000444                   MOV     #TEMP2,R2
2059  007432  105022                           CLRB    (R2)+           ; START CLEAN
2060  007434  112710  000252                   MOVB    #252,(R0)       ; LOAD THE LOCATIONS
2061  007440  112021                           MOVB    (R0)+,(R1)+     ; TEMP ---> TEMP1
2062  007442  105201                           INCB    R1              ; MAKE IT EVEN
2063  007444  111167  170770                   MOVB    (R1),TEMP       ; MORE 0'S INTO TEMP
2064  007450  105200                           INCB    R0              ; MAKE IT EVEN
2065  007452  112021                           MOVB    (R0)+,(R1)+     ; TEMP1 ---> TEMP2
2066  007454  124227  000252                   CMPB    -(R2),#252      ; CHECK IT
2067  007460  001003                           BNE     1$              ; FAILED
2068  007462  105767  170752                   TSTB    TEMP            ; CHECK IT
2069  007466  001404                           BEQ     2$              ; OK, CONTINUE
2070  007470                            1$:
2071  007470  012745  000211                   MOV     #211,-(R5)
2072  007474  005245                           INC     -(R5)
2073  007476  000000                           HALT                    ; INSTRUCTIONS FAILED IN MODE 2
2074
2075  007500  005741                    2$:    TST     -(R1)
2076  007502  005022                           CLR     (R2)+           ; START CLEAN
2077  007504  012740  125252                   MOV     #125252,-(R0)   ; LOAD LOCATIONS
2078  007510  012020                           MOV     (R0)+,(R0)+     ; TEMP ---> TEMP1
2079  007512  011067  170722                   MOV     (R0),TEMP       ; 0 ---> TEMP
2080  007516  012121                           MOV     (R1)+,(R1)+     ; 125252 ---> TEMP2
2081  007520  024227  125252                   CMP     -(R2),#125252   ; CHECK IT
2082  007524  001003                           BNE     EMODE2          ; FAILED
2083  007526  005767  170706                   TST     TEMP            ; CHECK IT
2084  007532  001404                           BEQ     MODE3           ; OK, CONTINUE
2085  007534                            EMODE2:
2086  007534  012745  000212                   MOV     #212,-(R5)
2087  007540  005245                           INC     -(R5)
2088  007542  000000                           HALT                    ; INSTRUCTIONS FAILED IN MODE 2
2089                                                                    ; OR WRONG SEQUENCE
```

```
2090                                    ;;*******************************************************************************
2091                                    ;*TEST: 56      CHECK MODE 3 USING THE MOVB AND MOV INSTRUCTIONS
2092                                    ;;*******************************************************************************
2093
2094  007544                            MODE3:
2095  007544  021527  000056                   CMP     (P5),#56
2096  007550  001066                           BNE     EMODE3          ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
2097  007552  005215                           INC     (R5)
2098  007554  012767  000440  170646           MOV     #TEMP,ADR       ; LOAD ADDRESSES
2099  007562  012767  000442  170642           MOV     #TEMP1,ADR1     ;
2100  007570  012767  000444  170636           MOV     #TEMP2,ADR2     ;
2101  007576  012700  000430                   MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
2102  007602  012701  000432                   MOV     #ADR1,R1        ;
2103  007606  105067  170632                   CLRB    TEMP2           ; START CLEAN
2104  007612  112767  000125  170620           MOVB    #125,TEMP       ;
2105  007620  113031                           MOVB    @(R0)+,@(R1)+   ; TEMP ---> TEMP1
2106  007622  113167  170612                   MOVB    @(R1)+,TEMP     ; TEMP2 ---> TEMP
2107  007626  113030                           MOVB    @(R0)+,@(R0)+   ; TEMP1 ---> TEMP2
2108  007630  122767  000125  170606           CMPB    #125,TEMP2      ; CHECK IT
2109  007636  001003                           BNE     1$              ; FAILED
2110  007640  105767  170574                   TSTB    TEMP            ; CHECK IT
2111  007644  001404                           BEQ     2$              ; OK, CONTINUE
2112  007646                            1$:
2113  007646  012745  000213                   MOV     #213,-(R5)
2114  007652  005245                           INC     -(R5)
2115  007654  000000                           HALT                    ; INSTRUCTIONS FAILED IN MODE 3
2116  007656  005067  170562            2$:    CLR     TEMP2           ; START CLEAN
2117  007662  012767  052525  170550           MOV     #52525,TEMP     ; LOAD LOCATIONS
2118  007670  012700  000430                   MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
2119  007674  012701  000432                   MOV     #ADR1,R1        ;
2120  007700  013030                           MOV     @(R0)+,@(R0)+   ; TEMP ---> TEMP1
2121  007702  013067  170532                   MOV     @(R0)+,TEMP     ; TEMP2 ---> TEMP
2122  007706  013131                           MOV     @(R1)+,@(R1)+   ; TEMP1 ---> TEMP2
2123  007710  022767  052525  170526           CMP     #52525,TEMP2    ; CHECK IT
2124  007716  001003                           BNE     EMODE3          ; FAILED
2125  007720  005767  170514                   TST     TEMP            ; CHECK IT
2126  007724  001404                           BEQ     MODE4           ; OK, CONTINUE
2127  007726                            EMODE3:
2128  007726  012745  000214                   MOV     #214,-(R5)
2129  007732  005245                           INC     -(R5)
2130  007734  000000                           HALT                    ; INSTRUCTIONS FAILED IN MODE 3
2131
2132
2133
2134                                    ;;*******************************************************************************
2135                                    ;*TEST: 57      CHECK MODE 4 USING THE MOVB AND MOV INSTRUCTIONS
2136                                    ;;*******************************************************************************
2137
2138  007736                            MODE4:
2139  007736  021527  000057                   CMP     (R5),#57
2140  007742  001120                           BNE     EMODE4          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
2141  007744  005215                           INC     (R5)
2142  007746  105067  170466                   CLRB    TEMP            ; START CLEAN
2143  007752  012700  000440                   MOV     #TEMP,R0        ; LOAD ADDRESSES
2144  007756  012701  000442                   MOV     #TEMP1,R1       ;
2145  007762  012702  000444                   MOV     #TEMP2,R2       ;
```

```
2146  007766  005202                          INC     R2                      ; ADJUST THE POINTER
2147  007770  021267  170451                  CMP     (R2),TEMP2+1
2148  007774  001404                          BEQ     1$
2149  007776  012745  000215                  MOV     #215,-(R5)
2150  010002  005245                          INC     -(R5)
2151  010004  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 4
2152  010006  112742  000252          2$:     MOVB    #252,-(R2)              ; LOAD TEMP2
2153  010012  005201                          INC     R1                      ; ADJUST THE POINTERS
2154  010014  005202                          INC     R2                      ;
2155  010016  114241                          MOVB    -(R2),-(R1)             ; TEMP2 ---> TEMP1
2156  010020  005200                          INC     R0                      ; ADJUST THE POINTERS
2157  010022  005202                          INC     R2                      ;
2158  010024  114042                          MOVB    -(R0),-(R2)             ; TEMP ---> TEMP2
2159  010026  105200                          INCB    R0                      ; ADJUST THE POINTERS
2160  010030  021067  170405                  CMP     (R0),TEMP+1
2161  010034  001404                          BEQ     2$
2162  010036  012745  000216                  MOV     #216,-(R5)
2163  010042  005245                          INC     -(R5)
2164  010044  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 4
2165  010046  105201                  2$:     INCB    R1                      ;
2166  010050  114140                          MOVB    -(R1),-(R0)             ; TEMP1 ---> TEMP
2167  010052  122767  000252  170360          CMPB    #252,TEMP               ; CHECK IT
2168  010060  001003                          BNE     3$                      ; FAILED
2169  010062  105767  170356                  TSTB    TEMP2                   ; CHECK IT
2170  010066  001404                          BEQ     4$                      ; OK, CONTINUE
2171  010070                          3$:
2172  010070  012745  000217                  MOV     #217,-(R5)
2173  010074  005245                          INC     -(R5)
2174  010076  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 4
2175  010100  005067  170334          4$:     CLR     TEMP                    ; START CLEAN
2176  010104  012700  000440                  MOV     #TEMP,R0                ; LOAD ADDRESSES
2177  010110  012701  000442                  MOV     #TEMP1,R1               ;
2178  010114  012702  000444                  MOV     #TEMP2,R2               ;
2179  010120  005722                          TST     (R2)+                   ; ADJUST THE POINTER
2180  010122  021267  170320                  CMP     (R2),TEMP2+2
2181  010126  001404                          BEQ     5$
2182  010130  012745  000220                  MOV     #220,-(R5)
2183  010134  005245                          INC     -(R5)
2184  010136  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 4
2185  010140  012742  125252          5$:     MOV     #125252,-(R2)           ; LOAD TEMP2
2186  010144  005721                          TST     (R1)+                   ; ADJUST THE POINTERS
2187  010146  005722                          TST     (R2)+                   ;
2188  010150  014241                          MOV     -(R2),-(R1)             ; TEMP2 ---> TEMP1
2189  010152  005720                          TST     (R0)+                   ; ADJUST POINTERS
2190  010154  005722                          TST     (R2)+                   ;
2191  010156  014042                          MOV     -(R0),-(R2)             ; TEMP ---> TEMP2
2192  010160  005720                          TST     (R0)+                   ; ADJUST THE POINTERS
2193  010162  005721                          TST     (R1)+                   ;
2194  010164  014140                          MOV     -(R1),-(R0)             ; TEMP1 ---> TEMP
2195  010166  022767  125252  170244          CMP     #125252,TEMP            ; CHECK IT
2196  010174  001003                          BNE     EMODE4                  ; FAILED
2197  010176  005767  170242                  TST     TEMP2                   ; CHECK IT
2198  010202  001404                          BEQ     MODE5                   ; OK, CONTINUE
2199  010204                          EMODE4:
2200  010204  012745  000221                  MOV     #221,-(R5)
2201  010210  005245                          INC     -(R5)
```

```
2202  010212  000000                        HALT                    ; INSTRUCTIONS FAILED IN MODE 4
2203                                                                  ; OR WRONG SEQUENCE
```

```
DVKAA8  MACY11 30(1046)  25-OCT-77  13:11  PAGE 60
DVKAA8.P11    25-OCT-77 13:09          T60     CHECK MODE 5 USING THE MOVB AND MOV INSTRUCTIONS                        SEQ 0060

                                               ;;*******************************************************************************
                                               ;*TEST: 60      CHECK MODE 5 USING THE MOVB AND MOV INSTRUCTIONS
                                               ;;*******************************************************************************

010214                                         MODE5:
010214  021527  000060                                 CMP     (PS),#60
010220  001105                                         BNE     EMODE5          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
010222  005215                                         INC     (R5)
010224  105067  170210                               CLRB    TEMP            ; START CLEAN
010230  012767  000440  170172                       MOV     #TEMP,ADR       ; LOAD ADDRESSES
010236  012767  000442  170166                       MOV     #TEMP1,ADR1     ;
010244  012767  000444  170162                       MOV     #TEMP2,ADR2     ;
010252  012700  000430                               MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
010256  012701  000432                               MOV     #ADR1,R1        ;
010262  012702  000434                               MOV     #ADR2,R2        ;
010266  005722                                       TST     (R2)+           ; ADJUST THE POINTER
010270  112752  000125                               MOVB    #125,@-(R2)     ; LOAD TEMP2
010274  022122                                       CMP     (R1)+,(R2)+     ; ADJUST THE POINTERS
010276  1152C1                                       MOVB    @-(R2),@-(R1)   ; TEMP2 ---> TEMP1
010300  0220C2                                       CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
010302  115052                                       MOVB    @-(R0),@-(R2)   ; TEMP ---> TEMP2
010304  022022                                       CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
010306  125052                                       CMPB    @-(R0),@-(R2)   ;CHECK IT
010310  001404                                       BEQ     1$
010312  012745  000222                               MOV     #222,-(R5)
010316  005245                                       INC     -(R5)
010320  000000                                       HALT
010322  022120                                 1$:   CMP     (R1)+,(R0)+     ; ADJUST THE POINTERS
010324  115150                                       MOVB    @-(R1),@-(R0)   ; TEMP1 ---> TEMP
010326  122767  000125  170104                       CMPB    #125,TEMP       ; CHECK IT
010334  001003                                       BNE     2$              ; FAILED
010336  105767  170102                               TSTB    TEMP2           ; CHECK IT
010342  001404                                       BEQ     3$              ; OK, CONTINUE
010344                                         2$:
010344  012745  000223                               MOV     #223,-(R5)
010350  005245                                       INC     -(R5)
010352  000000                                       HALT                    ; INSTRUCTIONS FAILED IN MODE 5
010354  005067  170060                         3$:   CLR     TEMP            ; START CLEAN
010360  012700  000430                               MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
010364  012701  000432                               MOV     #ADR1,R1        ;
010370  012702  000434                               MOV     #ADR2,R2        ;
010374  005722                                       TST     (R2)+           ; ADJUST THE POINTER
010376  012752  052525                               MOV     #52525,@-(R2)   ; LOAD TEMP2
010402  022122                                       CMP     (R1)+,(R2)+     ; ADJUST THE POINTERS
010404  015251                                       MOV     @-(R2),@-(R1)   ; TEMP2 ---> TEMP1
010406  022022                                       CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
010410  015052                                       MOV     @-(R0),@-(R2)   ; TEMP ---> TEMP2
010412  022021                                       CMP     (R0)+,(R1)+     ; ADJUST THE POINTERS
010414  015150                                       MOV     @-(R1),@-(R0)   ; TEMP1 ---> TEMP
010416  022767  052525  170014                       CMP     #52525,TEMP     ; CHECK IT
010424  001003                                       BNE     EMODE5          ; FAILED
010426  005767  170012                               TST     TEMP2           ; CHECK IT
010432  001404                                       BEQ     MODE6           ; OK, CONTINUE
010434                                         EMODE5:
010434  012745  000224                               MOV     #224,-(R5)
010440  005245                                       INC     -(R5)
```

```
2260  010442  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 5
2261                                                                  ; OR WRONG SEQUENCE
2262
2263
2264                                  ;;*************************************************************************
2265                                  ;;*TEST: 61      CHECK MODE 6 USING THE MOVB AND MOV INSTRUCTIONS
2266                                  ;;*************************************************************************
2267
2268  010444                          MODE6:
2269  010444  021527  000061                 CMP     (R5),#61
2270  010450  001055                         BNE     EMODE6          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
2271  010452  005215                         INC     (R5)
2272  010454  005067  167764                 CLR     TEMP2           ; START CLEAN
2273  010460  012700  000440                 MOV     #TEMP,R0        ; LOAD ADDRESSES
2274  010464  012701  000442                 MOV     #TEMP1,R1       ;
2275  010470  012702  000444                 MOV     #TEMP2,R2       ;
2276  010474  112760  000252  000000         MOVB    #252,0(R0)      ; LOAD TEMP (LOW BYTE)
2277  010502  112760  000252  000001         MOVB    #252,1(R0)      ; LOAD TEMP (HIGH BYTE)
2278  010510  022767  125252  167722         CMP     #125252,TEMP    ; CHECK IT
2279  010516  001012                         BNE     1$              ; FAILED
2280  010520  116062  000001  000000         MOVB    1(R0),0(R2)     ; TEMP(H) ---> TEMP2(L)
2281  010526  116160  000002  000005         MOVB    2(R1),5(R0)     ; TEMP2(L) ---> TEMP2(H)
2282  010534  022767  125252  167702         CMP     #125252,TEMP2   ; CHECK IT
2283  010542  001404                         BEQ     2$              ; OK, CONTINUE
2284  010544                          1$:
2285  010544  012745  000225                 MOV     #225,-(R5)
2286  010550  005245                         INC     -(R5)
2287  010552  000000                         HALT                    ; INSTRUCTIONS FAILED IN MODE 6
2288  010554  005067  167662          2$:    CLR     TEMP1           ; START CLEAN
2289  010560  012760  052525  000000         MOV     #52525,0(R0)    ; LOAD TEMP
2290  010566  016260  177774  000002         MOV     -4(R2),2(R0)    ; TEMP ---> TEMP1
2291  010574  022767  052525  167640         CMP     #52525,TEMP1    ; CHECK IT
2292  010602  001404                         BEQ     MODE7           ; OK, CONTINUE
2233  010604                          EMODE6:
2294  010604  012745  000226                 MOV     #226,-(R5)
2295  010610  005245                         INC     -(R5)
2296  010612  000000                         HALT                    ; INSTRUCTIONS FAILED IN MODE 6
2297                                                                  ; OR WRONG SEQUENCE
```

```
2298                                    ;;*******************************************************************************
2299                                    ;*TEST: 62       CHECK MODE 7 USING THE MOVB AND MOV INSTRUCTIONS
2300                                    ;;*******************************************************************************
2301
2302  010614                            MODE7:
2303  010614  021527  000062                   CMP     (PS),#62
2304  010620  001052                           BNE     EMODE7          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
2305  010622  005215                           INC     (R5)
2306  010624  005067  167612                   CLR     TEMP1           ; START CLEAN
2307  010630  012767  000440  167572           MOV     #TEMP,ADR       ; LOAD ADDRESSES
2308  010636  012767  000442  167566           MOV     #TEMP1,ADR1     ;
2309  010644  012767  000444  167562           MOV     #TEMP2,ADR2     ;
2310  010652  012700  000430                   MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
2311  010656  012701  000432                   MOV     #ADR1,R1        ;
2312  010662  012702  000434                   MOV     #ADR2,R2        ;
2313  010666  112770  000252  000000           MOVB    #252,@0(R0)     ; LOAD TEMP
2314  010674  117270  177774  000002           MOVB    @-4(R2),@2(R0)  ; TEMP ---> TEMP1
2315  010702  122767  000252  167532           CMPB    #252,TEMP1      ; CHECK IT
2316  010710  001404                           BEQ     1$              ; OK, CONTINUE
2317  010712  012745  000227                   MOV     #227,-(R5)
2318  010716  005245                           INC     -(R5)
2319  010720  000000                           HALT                    ; MODE 7 IS FAILING
2320  010722  012770  125252  000000  1$:      MOV     #125252,@0(R0)  ; LOAD TEMP
2321  010730  017270  177774  000002           MOV     @-4(R2),@2(R0)  ; TEMP ---> TEMP1
2322  010736  022767  125252  167476           CMP     #125252,TEMP1   ; CHECK IT
2323  010744  001404                           BEQ     TSTB1           ; OK, CONTINUE
2324  010746                            EMODE7:
2325  010746  012745  000230                   MOV     #230,-(R5)
2326  010752  005245                           INC     -(R5)
2327  010754  000000                           HALT                    ; INSTRUCTIONS FAILED IN MODE 7
2328                                                                    ; OR WRONG SEQUENCE
```

```
                                ;       CHECK BYTE INSTRUCTIONS, NOT DESTINATION MODE 0
                                ;       ----------------------------------------------
                                ;
                                ;
                                ;;*****************************************************************************
                                ;;TEST: 63      NEW INSTRUCTIONS USED IN THIS SECTION ARE TSTB, CLRB, MOVB
                                ;;*****************************************************************************
010756                          TSTB1:
010756  021527  000063                  CMP     (R5),#63
010762  001042                          BNE     ETSTB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF TEST
010764  005215                  2$:     INC     (R5)
010766  012700  000440                  MOV     #TEMP,R0        ; LOAD ADDRESSES
010772  012701  000442                  MOV     #TEMP1,R1       ;
010776  000277                          SCC
011000  105010                          CLRB    (R0)            ; CLEAR THE LOCATION
011002  004737  017230                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
011006  105710                          TSTB    (R0)            ; CHECK IT
011010  004737  017230                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
011014  112711  000377                  MOVB    #377,(R1)       ; LOAD THE LOCATION
011020  004737  017316                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
011024  105711                          TSTB    (R1)            ; CHECK IT
011026  004737  017316                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
011032  010002                          MOV     R0,R2           ; R2 IS NOW POINTING TO LOCATION TEMP
011034  112762  000200  000000          MOVB    #200,0(R2)      ; PLACE #200 IN LOCATION TEMP
011042  112241                          MOVB    (R2)+,-(R1)     ; MOVE #200 TO LOCATION TEMP+1
011044  026127  177777  100200          CMP     -1(R1),#100200  ; CHECK THE DATA IN LOCATION TEMP
011052  001404                          BEQ     4$
011054  012745  000231                  MOV     #231,-(R5)
011060  005245                          INC     -(R5)
011062  000000                          HALT                    ; MOVB INSTRUCTION FAILED
011064  020102                  4$:     CMP     R1,R2           ; CHECK THE REGISTERS FOR PROPER VALUE
011066  001404                          BEQ     CMPB1
011070                          ETSTB1:
011070  012745  000232                  MOV     #232,-(R5)
011074  005245                          INC     -(R5)
011076  000000                          HALT                    ; MOVB INSTRUCTION FAILED OR WRONG SEQUENCE
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 64      NEW INSTRUCTIONS USED IN THIS SECTION ARE CMPB, BISB
                                        ;;*******************************************************************************

011100                                  CMPB1:
011100  021527  000064                          CMP     (R5),#64
011104  001032                          1$:     BNE     ECMPB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
011106  005215                                  INC     (R5)
011110  012701  000444                          MOV     #TEMP2,R1
011114  012702  000440                          MOV     #TEMP,R2        ; LOAD ADDRESS
011120  012711  000077                          MOV     #77,(R1)        ; PLACE 77 IN LOCATION TEMP2
011124  112704  000377                          MOVB    #377,R4         ; R4 SHOULD CONTAIN #177777
011130  150412                                  BISB    R4,(R2)         ; LOAD LOCATION
011132  004737  017316                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
011136  120412                                  CMPB    R4,(R2)         ; CHECK COMPARE
011140  001404                                  BEQ     2$              ; CONTINUE IF OK
011142  012745  000233                          MOV     #233,-(R5)
011146  005245                                  INC     -(R5)
011150  000000                                  HALT                    ; BISB OR CMPB INSTRUCTION FAILED
011152  121112                          2$:     CMPB    (R1),(R2)       ; CHECK IT AGAIN
011154  100004                                  BPL     3$              ; CONTINUE IF OK
011156  012745  000234                          MOV     #234,-(R5)
011162  005245                                  INC     -(R5)
011164  000000                                  HALT                    ; CMPB INSTRUCTION FAILED (WRONG CC)
011166  121211                          3$:     CMPB    (R2),(R1)       ; ONCE MORE
011170  100404                                  BMI     BICB1           ; CONTINUE IF OK
011172                                  ECMPB1:
011172  012745  000235                          MOV     #235,-(R5)
011176  005245                                  INC     -(R5)
011200  000000                                  HALT


                                        ;;*******************************************************************************
                                        ;*TEST: 65      NEW INSTRUCTIONS USED IN THIS SECTION ARE BICB, BITB
                                        ;;*******************************************************************************

011202                                  BICB1:
011202  021527  000065                          CMP     (R5),#65
011206  001404                                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
011210  012745  000236                          MOV     #236,-(R5)
011214  005245                                  INC     -(R5)
011216  000000                                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
011220  005215                          2$:     INC     (R5)
011222  012703  000440                          MOV     #TEMP,R3        ; LOAD ADDRESS
011226  112713  000377                          MOVB    #377,(R3)       ; LOAD LOCATION
011232  012700  000442                          MOV     #TEMP1,R0       ;PLACE THE ADDRESS OF LOCATION TEMP1 IN R0
011236  010001                                  MOV     R0,R1           ;AND R1
011240  112721  000252                          MOVB    #252,(R1)+      ; PLACE #252 IN TEMP1
011244  000277                                  SCC
011246  146013  000000                          BICB    0(R0),(R3)      ; CLEAR EVERY OTHER BIT
011252  004737  017146                          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
011256  136113  177777                          BITB    -1(R1),(R3)     ; CHECK IT
011262  001404                                  BEQ     4$              ; CONTINUE IF OK
011264  012745  000237                          MOV     #237,-(R5)
011270  005245                                  INC     -(R5)
```

```
                        T65     NEW INSTRUCTIONS USED IN THIS SECTION ARE BICB, BITB                    SEQ 0065

 2427  011272  000000                        HALT                    ; BICB OR BITB INSTRUCTION FAILED
 2428  011274  132713  000125        4$:     BITB    #125,(R3)       ; CHECK IT
 2429  011300  004737  017146                JSR     PC,@#$CC1       ; CHECK FOR CC = 1
 2430  011304  154113                        BISB    -(R1),(R3)      ; SET THE BITS THAT WERE CLEARED
 2431  011306  100404                        BMI     6$              ; CONTINUE IF OK
 2432  011310  012745  000240                MOV     #240,-(R5)
 2433  011314  005245                        INC     -(R5)
 2434  011316  000000                        HALT                    ; BITB OR BISB INSTRUCTION FAILED
 2435  011320  012746  000177        6$:     MOV     #177,-(SP)      ; STORE #177 ON THE STACK
 2436  011324  142613                        BICB    (SP)+,(R3)      ; CLEAR ALL THE BITS EXCEPT SIGN BIT
 2437  011326  004737  017336                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 2438  011332  132713  000377                BITB    #377,(R3)       ; CHECK IT
 2439  011336  004737  017336                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 2440  011342  010300                        MOV     R3,R0           ; PLACE THE ADDRESS OF LOCATION TEMP IN R0
 2441  011344  012710  000442                MOV     #TEMP1,(R0)     ; PLACE THE ADDRESS OF LOCATION TEMP1 IN TEMP
 2442  011350  012730  000377                MOV     #377,@(R0)+     ; WRITE A 377 IN LOCATION TEMP1
 2443  011354  000263                        SEVC                    ; SET V & C BITS
 2444  011356  145070  000000                BICB    @-(R0),@(R0)    ; BIT CLEAR THE CONTENTS
 2445                                                                ; OF TEMP1 TO THE CONTENTS OF TEMP1
 2446  011362  004737  017252                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
 2447  011366  022027  000442                CMP     (R0)+,#TEMP1    ; MAKE SURE THAT (R0) IS POINTING TO LOCATION TEMP1
 2448  011372  001404                        BEQ     8$
 2449  011374  012745  000241                MOV     #241,-(R5)
 2450  011400  005245                        INC     -(R5)
 2451  011402  000000                        HALT                    ; BICB OR CMP INSTRUCTION FAILED IN THE SPECIFIC MODE
 2452  011404  005750          8$:     TST     @-(R0)          ; TEST LOCATION TEMP1
 2453  011406  001404                        BEQ     10$
 2454  011410  012745  000242                MOV     #242,-(R5)
 2455  011414  005245                        INC     -(R5)
 2456  011416  000000                        HALT                    ; BICB INSTRUCTION FAILED
 2457  011420  000257          10$:    CCC
 2458  011422  141010                        BICB    (R0),(R0)       ; CLEAR THE LOCATION TEMP
 2459  011424  004737  017230                JSR     PC,@#$CC4       ; CHECK FOR CC = 4
```

```
2460
2461                                    ;;*******************************************************************************
2462                                    ;;*TEST: 66      NEW INSTRUCTIONS USED IN THIS SECTION ARE INCB, DECB
2463                                    ;;*******************************************************************************
2464   011430                           INCB1:
2465   011430  021527  000066                  CMP     (R5),#66
2466   011434  001067                          BNE     EINCB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
2467   011436  005215                  1$:     INC     (R5)
2468   011440  012704  000440                 MOV     #TEMP,R4        ; LOAD ADDRESS
2469   011444  112714  000177                 MOVB    #177,(R4)       ; TEMP LOCATION=177
2470   011450  000261                          SEC
2471   011452  105214                          INCB    (R4)            ; ADD ONES INTO LOCATION
2472   011454  004737  017400                 JSR     PC,@#$CC13      ; CHECK FOR CC = 13
2473   011460  012714  000376                 MOV     #376,(R4)
2474   011464  012700  017336                 MOV     #$CC11,R0       ; MAKE R0 POINT TO CHECKING ROUTINE FOR CC = 11
2475   011470  105224                          INCB    (R4)+           ;
2476   011472  004720                          JSR     PC,(R0)+        ; CHECK FOR CC = 11
2477   011474  105744                          TSTB    -(R4)           ; DECREMENT R4 BY 1
2478   011476  005746                          TST     -(SP)           ; AND SP BY 2
2479   011500  010426                          MOV     R4,(SP)+        ; PLACE THE ADDRESS OF TEMP ON THE STACK
2480   011502  000241                          CLC                     ; CLEAR C BIT
2481   011504  105256                          INCB    @-(SP)          ; INCREMENT THE CONTENTS OF LOCATION TEMP
2482   011506  004737  017230                 JSR     PC,@#$CC4       ; CHECK FOR CC = 4
2483   011512  123634                          CMPB    @(SP)+,@(R4)+   ; RESTORE STACK POINTER
2484   011514  000261                          SEC                     ; SET C BIT
2485   011516  105264  177777                  INCB    -1(R4)
2486   011522  004737  017146                 JSR     PC,@#$CC1       ; CHECK FOR CC = 1
2487   011526  124427  000001                 CMPB    -(R4),#1        ; CHECK IT
2488   011532  001404                          BEQ     2$              ; CONTINUE IF OK
2489   011534  012745  000243                 MOV     #243,-(R5)
2490   011540  005245                          INC     -(R5)
2491   011542  000000                          HALT                    ; INCB INSTRUCTION FAILED
2492   011544  000261                  2$:     SEC
2493   011546  105314                          DECB    (R4)            ; SUBTRACT ONES FROM LOCATION
2494   011550  004737  017252                 JSR     PC,@#$CC5       ; CHECK FOR CC = 5
2495   011554  105324                          DECB    (R4)+
2496   011556  004740                          JSR     PC,-(R0)        ; CHECK FOR CC = 11
2497   011560  112764  000200  177777         MOVB    #200,-1(R4)
2498   011566  105344                          DECB    -(R4)           ;
2499   011570  004760  177650                 JSR     PC,$CC3-$CC11(R0); CHECK FOR CC = 3
2500   011574  105364  000000                 DECB    0(R4)           ;
2501   011600  004737  017146                 JSR     PC,@#$CC1       ; CHECK FOR CC = 1
2502   011604  126427  000000  000176         CMPB    0(R4),#176
2503   011612  001404                          BEQ     COMB1
2504   011614                           EINCB1:
2505   011614  012745  000244                 MOV     #244,-(R5)
2506   011620  005245                          INC     -(R5)
2507   011622  000000                          HALT                    ; DECB INSTRUCTION FAILED OR SEQUENCE ERROR
2508
2509
2510
2511
2512
2513
2514
2515
```

```
2516
2517                                    ;;*******************************************************************************
2518                                    ;*TEST: 67      NEW INSTRUCTION IN THIS SECTION IS COMB
2519                                    ;;*******************************************************************************
2520
2521  011624                            COMB1:
2522  011624  021527  000067                   CMP     (R5),#67
2523  011630  001404                           BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
2524  011632  012745  000245                   MOV     #245,-(R5)
2525  011636  005245                           INC     -(R5)
2526  011640  000000                           HALT                            ; TEST IS IN WRONG SEQUENCE
2527  011642  005215                    1$:    INC     (R5)
2528  011644  012703  000440                   MOV     #TEMP,R3                ; LOAD ADDRESS
2529  011650  012704  000442                   MOV     #TEMP1,R4
2530  011654  012714  000252                   MOV     #252,(R4)
2531  011660  112413                           MOVB    (R4)+,(R3)              ; LOAD EVERY OTHER BIT
2532  011662  000277                           SCC
2533  011664  105113                           COMB    (R3)                    ; 1'S COMPLEMENT
2534  011666  004737  017146                   JSR     PC,@#$CC1               ; CHECK FOR CC = 1
2535  011672  122713  000125                   CMPB    #125,(R3)               ; CHECK IT
2536  011676  001404                           BEQ     2$                      ; CONTINUE IF OK
2537  011700  012745  000246                   MOV     #246,-(R5)
2538  011704  005245                           INC     -(R5)
2539  011706  000000                           HALT                            ; COMB INSTRUCTION FAILED
2540  011710  000277                    2$:    SCC
2541  011712  105113                           COMB    (R3)                    ; COMPLEMENT BACK
2542  011714  004737  017336                   JSR     PC,@#$CC11              ; CHECK FOR CC = 11
2543  011720  010400                           MOV     R4,R0
2544  011722  126013  177777                   CMPB    -1(R0),(R3)             ; CHECK IT
2545  011726  001404                           BEQ     3$                      ; CONTINUE IF OK
2546  011730  012745  000247                   MOV     #247,-(R5)
2547  011734  005245                           INC     -(R5)
2548  011736  000000                           HALT                            ; COMB INSTRUCTION FAILED
2549  011740  112724  000377            3$:   MOVB    #377,(R4)+
2550  011744  114413                           MOVB    -(R4),(R3)              ; PLACE #377 IN (R3)
2551  011746  000277                           SCC
2552  011750  105113                           COMB    (R3)
2553  011752  004737  017252                   JSR     PC,@#$CC5               ; CHECK FOR CC = 5
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 70       NEW INSTRUCTION IN THIS SECTION IS NEGB
                                        ;;*******************************************************************************

011756                                  NEGB1:
011756  021527  000070                          CMP     (R5),#70
011762  001027                                 BNE     ENEGB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
011764  005215                          1$:     INC     (R5)
011766  012700  000440                          MOV     #TEMP,R0        ; LOAD ADDRESS
011772  112710  000001                          MOVB    #1,(R0)         ; LOAD THE LOCATION
011776  105410                                 NEGB    (R0)            ; 2'S COMPLEMENT
012000  004737  017336                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
012004  122710  000377                          CMPB    #377,(R0)       ; CHECK IT
012010  001404                                 BEQ     2$              ; CONTINUE IF OK
012012  012745  000250                          MOV     #250,-(R5)
012016  005245                                 INC     -(R5)
012020  000000                                 HALT                    ; NEGB INSTRUCTION FAILED
012022  012710  000200                  2$:     MOV     #200,(R0)
012026  105410                                 NEGB    (R0)            ; 2'S COMPLEMENT
012030  004737  017400                          JSR     PC,@#$CC13      ; CHECK FOR CC = 13
012034  122710  000200                          CMPB    #200,(R0)       ; CHECK IT
012040  001404                                 BEQ     ROLB1           ; CONTINUE IF OK
012042                                  ENEGB1:
012042  012745  000251                          MOV     #251,-(R5)
012046  005245                                 INC     -(R5)
012050  000000                                 HALT                    ; WRONG RESULT AT TEMP OR WRONG SEQUENCE


                                        ;;*******************************************************************************
                                        ;*TEST: 71       NEW INSTRUCTION IN THIS SECTION IS ROLB
                                        ;;*******************************************************************************

012052                                  ROLB1:
012052  021527  000071                          CMP     (R5),#71
012056  001030                                 BNE     EROLB1          ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
012060  005215                                 INC     (R5)
012062  012701  000442                          MOV     #TEMP1,R1       ; LOAD ADDRESS
012066  112711  000040                          MOVB    #40,(R1)        ; LOAD LOCATION
012072  000257                                 CCC                     ; CLEAR FLAGS
012074  106111                                 ROLB    (R1)            ; SHIFT
012076  106111                                 ROLB    (R1)            ;
012100  004737  017360                          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
012104  122711  000200                          CMPB    #200,(R1)       ; CHECK IT
012110  001404                                 BEQ     1$              ; CONTINUE IF OK
012112  012745  000252                          MOV     #252,-(R5)
012116  005245                                 INC     -(R5)
012120  000000                                 HALT                    ; ROLB INSTRUCTION FAILED
012122  106111                          1$:     ROLB    (R1)            ; SHIFT
012124  004737  017274                          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
012130  106111                                 ROLB    (R1)            ; SHIFT
012132  122711  000001                          CMPB    #1,(R1)         ; CHECK IT
012136  001404                                 BEQ     RORB1           ; CONTINUE IF OK
012140                                  EROLB1:
012140  012745  000253                          MOV     #253,-(R5)
012144  005245                                 INC     -(R5)
```

```
  2610  012146  000000                        HALT                    ; WRONG RESULT AT TEMP1 OR WRONG SEQUENCE
```

```
                                        T72     NEW INSTRUCTION IN THIS SECTION IS RORB                                 SEQ 0070

2611                                    ;;*************************************************************************
2612                                    ;*TEST: 72      NEW INSTRUCTION IN THIS SECTION IS RORB
2613                                    ;;*************************************************************************
2614
2615  012150                            RORB1:
2616  012150  021527  000072                   CMP     (R5),#72
2617  012154  001030                           BNE     ERORB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
2618  012156  005215                           INC     (R5)
2619  012160  012702  000442                   MOV     #TEMP1,R2       ; LOAD ADDRESS
2620  012164  112712  000004                   MOVB    #4,(R2)         ; LOAD LOCATION
2621  012170  000257                           CCC                     ; CLEAR FLAGS
2622  012172  106012                           RORB    (R2)            ; SHIFT
2623  012174  106012                           RORB    (R2)            ;
2624  012176  122712  000001                   CMPB    #1,(R2)         ; CHECK IT
2625  012202  001404                           BEQ     1$              ; CONTINUE IF OK
2626  012204  012745  000254                   MOV     #254,-(R5)
2627  012210  005245                           INC     -(R5)
2628  012212  000000                           HALT                    ; RORB INSTRUCTION FAILED
2629  012214  106012                    1$:    RORB    (R2)            ; SHIFT
2630  012216  004737  017274                   JSR     PC,@#$CC7       ; CHECK FOR CC = 7
2631  012222  106012                           RORB    (R2)            ; SHIFT
2632  012224  004737  017360                   JSR     PC,@#$CC12      ; CHECK FOR CC = 12
2633  012230  122712  000200                   CMPB    #200,(R2)       ; CHECK IT
2634  012234  001404                           BEQ     ASLB1           ; CONTINUE IF OK
2635  012236                            ERORB1:
2636  012236  012745  000255                   MOV     #255,-(R5)
2637  012242  005245                           INC     -(R5)
2638  012244  000000                           HALT                    ; WRONG RESULT AT TEMP1 OR WRONG SEQUENCE
2639
2640
2641
2642                                    ;;*************************************************************************
2643                                    ;*TEST: 73      NEW INSTRUCTION IN THIS SECTION IS ASLB
2644                                    ;;*************************************************************************
2645
2646  012246                            ASLB1:
2647  012246  021527  000073                   CMP     (R5),#73
2648  012252  001404                           BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
2649  012254  012745  000256                   MOV     #256,-(R5)
2650  012260  005245                           INC     -(R5)
2651  012262  000000                           HALT                    ; PROGRAM IS IN WRONG SEQUENCE
2652  012264  005215                    2$:    INC     (R5)
2653  012266  012703  000442                   MOV     #TEMP1,R3       ; LOAD ADDRESS
2654  012272  112713  000040                   MOVB    #40,(R3)        ; LOAD LOCATION
2655  012276  000257                           CCC                     ; CLEAR FLAGS
2656  012300  106313                           ASLB    (R3)            ; SHIFT
2657  012302  106313                           ASLB    (R3)            ;
2658  012304  004737  017360                   JSR     PC,@#$CC12      ; CHECK FOR CC = 12
2659  012310  122713  000200                   CMPB    #200,(R3)       ; CHECK IT
2660  012314  001404                           BEQ     4$              ; CONTINUE IF OK
2661  012316  012745  000257                   MOV     #257,-(R5)
2662  012322  005245                           INC     -(R5)
2663  012324  000000                           HALT                    ; ASLB INSTRUCTION FAILED
2664  012326  106313                    4$:    ASLB    (R3)            ; SHIFT
2665  012330  004737  017274                   JSR     PC,@#$CC7       ; CHECK FOR CC = 7
2666  012334  106313                           ASLB    (R3)            ; SHIFT
```

```
  2667  012336  004737  017230          173     NEW INSTRUCTION IN THIS SECTION IS ASLB          SEQ 0071
                                                JSR     PC,@#$CC4       ; CHECK FOR CC = 4
```

```
2668
2669
2670
2671
2672  012342
2673  012342  021527  000074
2674  012346  001040
2675  012350  005215
2676  012352  012704  000442
2677  012356  012703  000444
2678  012362  112714  000004
2679  012366  000257
2680  012370  106214
2681  012372  106214
2682  012374  122714  000001
2683  012400  001404
2684  012402  012745  000260
2685  012406  005245
2686  012410  000000
2687  012412  106214
2688  012414  004737  017274
2689  012420  106214
2690  012422  004737  017230
2691  012426  112713  000202
2692  012432  106213
2693  012434  106213
2694  012436  004737  017336
2695  012442  122713  000340
2696  012446  001404
2697  012450
2698  012450  012745  000261
2699  012454  005245
2700  012456  000000
2701
2702
2703
2704
2705
2706
2707
2708  012460
2709  012460  021527  000075
2710  012464  001404
2711  012466  012745  000262
2712  012472  005245
2713  012474  000000
2714  012476  005215
2715  012500  012700  000444
2716  012504  105010
2717  012506  000257
2718  012510  105510
2719  012512  004737  017230
2720  012516  000261
2721  012520  105510
2722  012522  000261
2723  012524  105510
```

```
;;*****************************************************************************
;*TEST: 74      NEW INSTRUCTION IN THIS SECTION IS ASRB
;;*****************************************************************************

ASRB1:
        CMP     (P5),#74
        BNE     EASRB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
1$:     INC     (R5)
        MOV     #TEMP1,R4       ; LOAD ADDRESSES
        MOV     #TEMP2,R3       ;
        MOVB    #4,(R4)         ; LOAD LOCATION
        CCC                     ; CLEAR FLAGS
        ASRB    (R4)            ; SHIFT
        ASRB    (R4)            ;
        CMPB    #1,(R4)         ; CHECK IT
        BEQ     2$              ; CONTINUE IF OK
        MOV     #260,-(R5)
        INC     -(R5)
        HALT                    ; ASRB INSTRUCTION FAILED
2$:     ASRB    (R4)            ; SHIFT
        JSR     PC,@#$CC7       ; CHECK FOR CC = 7
        ASRB    (R4)            ; SHIFT
        JSR     PC,@#$CC4       ; CHECK FOR CC = 4
        MOVB    #202,(R3)       ; LOAD LOCATION
        ASRB    (R3)            ; SHIFT
        ASRB    (R3)            ;
        JSR     PC,@#$CC11      ; CHECK FOR CC = 11
        CMPB    #340,(R3)       ; CHECK IT
        BEQ     ADCB1           ; CONTINUE IF OK
EASRB1:
        MOV     #261,-(R5)
        INC     -(R5)
        HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE


;;*****************************************************************************
;*TEST: 75      NEW INSTRUCTION IN THIS SECTION IS ADCB
;;*****************************************************************************

ADCB1:
        CMP     (R5),#75
        BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
        MOV     #262,-(R5)
        INC     -(R5)
        HALT                    ; PROGRAM IS IN WRONG SEQUENCE
2$:     INC     (R5)
        MOV     #TEMP2,R0       ; LOAD ADDRESS
        CLRB    (R0)            ; CLEAR THE LOCATION
        CCC                     ; CLEAR FLAGS
        ADCB    (R0)            ; ADD C BIT = 0
        JSR     PC,@#$CC4       ; CHECK FOR CC = 4
        SEC                     ; C=1
        ADCB    (R0)            ; ADD C BIT=1
        SEC                     ; C=1
        ADCB    (R0)            ; AGAIN
```

T75 NEW INSTRUCTION IN THIS SECTION IS ADCB SEQ 0073

```
2724  012526  004737  017126          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
2725  012532  122710  000002          CMPB    #2,(R0)         ; CHECK IT
2726  012536  001404                  BEQ     4$              ; CONTINUE IF OK
2727  012540  012745  000263          MOV     #263,-(R5)
2728  012544  005245                  INC     -(R5)
2729  012546  000000                  HALT                    ; ADCB INSTRUCTION FAILED
2730  012550  112710  000177    4$:   MOVB    #177,(R0)       ; LOAD LARGEST POSITVE BYTE
2731  012554  000261                  SEC                     ; C=1
2732  012556  105510                  ADCB    (R0)            ; ADD C BIT=1
2733  012560  004737  017360          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
2734  012564  122710  000200          CMPB    #200,(R0)       ; CHECK IT
2735  012570  001404                  BEQ     6$              ; CONTINUE IF OK
2736  012572  012745  000264          MOV     #264,-(R5)
2737  012576  005245                  INC     -(R5)
2738  012600  000000                  HALT                    ; ADCB INSTRUCTION FAILED
2739  012602  112710  000377    6$:   MOVB    #377,(R0)       ; LOAD -1
2740  012606  000261                  SEC                     ; C=1
2741  012610  105510                  ADCB    (R0)            ; ADD C BIT=1
2742  012612  004737  017252          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

```
2743                                            ;;*************************************************************************
2744                                            ;*TEST: 76      NEW INSTRUCTION IN THIS SECTION IS SBCB
2745                                            ;;*************************************************************************
2746
2747   012616                                   SBCB1:
2748   012616  021527  000076                          CMP     (P5),#76
2749   012622  001404                                  BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
2750   012624  012745  000265                          MOV     #265,-(R5)
2751   012630  005245                                  INC     -(R5)
2752   012632  000000                                  HALT                    ; TEST IS IN WRONG SEQUENCE
2753   012634  005215                          1$:     INC     (R5)
2754   012636  012701  000444                          MOV     #TEMP2,R1       ; LOAD ADDRESS
2755   012642  112711  000003                          MOVB    #3,(R1)         ; LOAD LOCATION
2756   012646  000257                                  CCC                     ; CLEAR FLAGS
2757   012650  105611                                  SBCB    (R1)            ; SUBTRACT C BIT=0
2758   012652  004737  017126                          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
2759   012656  122711  000003                          CMPB    #3,(R1)         ; CHECK IT
2760   012662  001404                                  BEQ     2$              ; CONTINUE IF OK
2761   012664  012745  000266                          MOV     #266,-(R5)
2762   012670  005245                                  INC     -(R5)
2763   012672  000000                                  HALT                    ; SBCB INSTRUCTION FAILED
2764   012674  000261                          2$:     SEC                     ; C=1
2765   012676  105611                                  SBCB    (R1)            ; SUBTRACT C BIT=1
2766   012700  000261                                  SEC                     ; C=1
2767   012702  105611                                  SBCB    (R1)            ;
2768   012704  004737  017126                          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
2769   012710  122711  000001                          CMPB    #1,(R1)         ; CHECK IT
2770   012714  001404                                  BEQ     3$              ; CONTINUE IF OK
2771   012716  012745  000267                          MOV     #267,-(R5)
2772   012722  005245                                  INC     -(R5)
2773   012724  000000                                  HALT                    ; SBCB INSTRUCTION FAILED
2774   012726  000261                          3$:     SEC                     ; C=1
2775   012730  105611                                  SBCB    (R1)            ; SUBTRACT C BIT=1
2776   012732  004737  017230                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
2777   012736  000261                                  SEC                     ; C=1
2778   012740  105611                                  SBCB    (R1)            ; SUBTRACT C BIT = 1
2779   012742  004737  017336                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
2780   012746  122711  000377                          CMPB    #377,(R1)       ; CHECK IT
2781   012752  001404                                  BEQ     4$              ; CONTINUE IF OK
2782   012754  012745  000270                          MOV     #270,-(R5)
2783   012760  005245                                  INC     -(R5)
2784   012762  000000                                  HALT                    ; SBCB INSTRUCTION FAILED
2785   012764  112711  000200                  4$:     MOVB    #200,(R1)       ; LOAD R1
```

```
  2786  012770  000261                        SEC                     ; C=1
  2787  012772  105611                        SBCB    (R1)            ; SUBTRACT C BIT = 1
  2788  012774  004737  017166                JSR     PC,@#SCC2       ; CHECK FOR CC = 2
```

```
                                ;               CHECK WORD INSTRUCTIONS, NOT DESTINATION MODE 0
                                ;               -----------------------------------------------




                                ;;*************************************************************************
                                ;*TEST: 77       NEW INSTRUCTIONS USED IN THIS SECTION ARE TST, CLR, MOV
                                ;;*************************************************************************

013000                          TST1:
013000  021527  000077                  CMP     (R5),#77
013004  001404                          BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
013006  012745  000271                  MOV     #271,-(R5)
013012  005245                          INC     -(R5)
013014  000000                          HALT                            ; TEST IS IN A WRONG SEQUENCE
013016  005215                  1$:     INC     (R5)
013020  012701  000440                  MOV     #TEMP,R1                ; LOAD ADDRESSES
013024  012700  000442                  MOV     #TEMP1,R0               ;
013030  000277                          SCC
013032  005010                          CLR     (R0)                    ; CLEAR THE LOCATION
013034  004737  017230                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
013040  005720                          TST     (R0)+                   ; CHECK IT
013042  004737  017230                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
013046  010040                          MOV     R0,-(R0)
013050  012730  177777                  MOV     #177777,@(R0)+
013054  017011  177776                  MOV     @-2(R0),(R1)            ; LOAD THE LOCATION
013060  004737  017316                  JSR     PC,@#$CC10              ; CHECK FOR CC = 10
013064  005711                          TST     (R1)                    ; CHECK IT
013066  004737  017316                  JSR     PC,@#$CC10              ; CHECK FOR CC = 10
```

```
                        ;;*******************************************************************************
                        ;*TEST: 100     NEW INSTRUCTIONS USED IN THIS SECTION ARE CMP, BIS
                        ;;*******************************************************************************
013072                  CMP1:
013072  021527  000100          CMP     (PS),#100
013076  001113                  BNE     ECMP1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE EST
013100  005215          1$:     INC     (R5)
013102  012702  000442          MOV     #TEMP1,R2       ; LOAD ADDRESS
013106  012700  000440          MOV     #TEMP,R0        ; PLACE THE ADDRESS OF TEMP IN R0
013112  012720  177777          MOV     #177777,(R0)+   ; PLACE #177777 IN LOCATION TEMP AND INC. R0 BY 2
013116  054012                  BIS     -(R0),(R2)      ; LOAD LOCATION
013120  004737  017316          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
013124  022227  177777          CMP     (R2)+,#177777   ; CHECK COMPARE
013130  001404                  BEQ     2$              ; CONTINUE IF OK
013132  012745  000272          MOV     #272,-(R5)
013136  005245                  INC     -(R5)
013140  000000                  HALT                    ; CMP OR BIS INSTRUCTION FAILED
013142  020227  000444  2$:     CMP     R2,#TEMP1+2     ; CHECK R2 TO CONTAIN ADDRESS OF TEMP1+2
013146  001404                  BEQ     3$
013150  012745  000273          MOV     #273,-(R5)
013154  005245                  INC     -(R5)
013156  000000                  HALT                    ; NO AUTO INCREMENT
013160  022742  000077  3$:     CMP     #77,-(R2)       ; CHECK IT AGAIN
013164  004737  017146          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
013170  022722  077777          CMP     #77777,(R2)+
013174  004737  017400          JSR     PC,@#$CC13      ; CHECK FOR CC = 13
013200  024227  077777          CMP     -(R2),#77777    ; ONCE MORE
013204  004737  017316          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
013210  012767  052525  165226  MOV     #52525,TEMP2    ; SET EVERY OTHER BIT IN TEMP2
013216  012767  000444  165216  MOV     #TEMP2,TEMP1    ; PLACE THE ADDRESS OF TEMP2 IN LOCATION TEMP1
013224  012704  000430          MOV     #ADR,R4
013230  012714  000432          MOV     #ADR1,(R4)      ; PLACE THE ADDRESS OF ADR1 IN ADR POINTED BY R4
013234  012734  125252          MOV     #125252,@(R4)+  ; PLACE THE #125252 IN LOCATION ADR1
013240  057432  177776          BIS     @-2(R4),@(R2)+  ; SET EVERY OTHER BIT AT LOCATION TEMP2
                                                        ; AND INCREMENT R2 BY 2
013244  010200                  MOV     R2,R0           ; PLACE ADDRESS OF TEMP2 IN R0
013246  026027  177777          CMP     @-(R0),#177777  ; TEMP2 SHOULD CONTAIN ALL 1'S
013252  001404                  BEQ     4$
013254  012745  000274          MOV     #274,-(R5)
013260  005245                  INC     -(R5)
013262  000000                  HALT                    ; CMP OR BIS INSTUCTIONS FAILED IN MODES OTHER THAN 0
013264  020227  000444  4$:     CMP     R2,#TEMP1+2     ; R2 SHOULD CONTAIN THE ADDRESS FOR TEMP2
                                                        ; I.E. TEMP1+2
013270  001404                  BEQ     5$
013272  012745  000275          MOV     #275,-(R5)
013276  005245                  INC     -(R5)
013300  000000                  HALT                    ; MODE 5 IS FAILING
013302  005040          5$:     CLR     -(R0)           ; PLACE A 0 IN LOCATION TEMP
013304  010067  165134          MOV     R0,TEMP2        ; PLACE ADDRESS OF TEMP IN LOCATION TEMP2
013310  022020                  CMP     (R0)+,(R0)+     ; BUMP R0 BY 4
013312  055070  000002          BIS     @-(R0),@2(R0)   ; PLACE THE CONTENTS OF LOCATION TEMP2 AT TEMP
013316  022767  000440  165114  CMP     #TEMP,TEMP      ; LOCATION TEMP SHOULD CONTAIN ITS OWN ADDRESS
013324  001404                  BEQ     BIC1
013326                  ECMP1:
013326  012745  000276          MOV     #276,-(R5)
```

```
2881  013332  005245                        INC     -(R5)
2882  013334  000000                        HALT                    ; CMP OR BIS INSTRUCTIONS FAILED OR WRONG
2883                                                                  ; SEQUENCE COUNTER
2884
2885
2886
2887                                  ;;*****************************************************************************
2888                                  ;*TEST: 101     NEW INSTRUCTIONS USED IN THIS SECTION ARE BIC, BIT
2889                                  ;;*****************************************************************************
2890
2891  013336                          BIC1:
2892  013336  021527  000101                CMP     (R5),#101
2893  013342  001122                        BNE     EBIC1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
2894  013344  005215                        INC     (R5)
2895  013346  012703  000440                MOV     #TEMP,R3        ; LOAD ADDRESS
2896  013352  012713  177777                MOV     #177777,(R3)    ; LOAD LOCATION
2897  013356  012704  000430                MOV     #ADR,R4         ; PLACE THE ADDRESS OF ADR IN R4
2898  013362  012714  000432                MOV     #ADR1,(R4)      ; PLACE THE ADDRESS OF ADR1 IN ADR
2899  013366  011334                        MOV     (R3),@(R4)+     ; LOAD LOCATION ADR1 WITH #177777
2900  013370  012700  000442                MOV     #TEMP1,R0       ; PLACE THE ADDRESS OF TEMP1 IN R0
2901  013374  012710  125252                MOV     #125252,(R0)    ; SET EVERY OTHER BIT AT LOCATION TEMP1
2902  013400  000277                        SCC
2903  013402  042013                        BIC     (R0)+,(R3)      ; CLEAR EVERY OTHER BIT
2904  013404  004737  017146                JSR     PC,@#$CC1       ; CHECK FOR CC = 1
2905  013410  034013                        BIT     -(R0),(R3)      ; CHECK IT
2906  013412  001404                        BEQ     1$              ; CONTINUE IF OK
2907  013414  012745  000277                MOV     #277,-(R5)
2908  013420  005245                        INC     -(R5)
2909  013422  000000                        HALT                    ; BIC OR BIT INSTRUCTION FAILED
2910  013424  032713  052525          1$:   BIT     #52525,(R3)     ; CHECK IT
2911  013430  004737  017146                JSR     PC,@#$CC1       ; CHECK FOR CC = 1
2912  013434  056013  000000                BIS     0(R0),(R3)      ; SET THE BITS THAT WERE CLEARED
2913  013440  100404                        BMI     2$              ; CONTINUE IF OK
2914  013442  012745  000300                MOV     #300,-(R5)
2915  013446  005245                        INC     -(R5)
2916  013450  000000                        HALT                    ; BIT OR BIS INSTRUCTION FAILED
2917  013452  012720  077777          2$:   MOV     #77777,(R0)+    ; SET ALL THE BITS AT LOCATION TEMP1 EXCEPT SIGN BIT
2918  013456  010002                        MOV     R0,R2
2919  013460  046213  177776                BIC     -2(R2),(R3)     ; TRY CLEARING THE OTHER BITS
2920  013464  004737  017336                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
2921  013470  020027  000444                CMP     R0,#TEMP1+2     ; R0 SHOULD CONTAIN THE ADDRESS OF TEMP1+2
2922  013474  001404                        BEQ     3$
2923  013476  012745  000301                MOV     #301,-(R5)
2924  013502  005245                        INC     -(R5)
2925  013504  000000                        HALT
2926  013506  010020          3$:   MOV     R0,(R0)+        ; PLACE THE ADDRES OF LOCATION TEMP2 IN TEMP2
2927  013510  000263                        SEVC                    ; SET V & C BITS
2928  013512  045000                        BIC     @-(R0),R0       ; CLEAR R0
2929  013514  004737  017252                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
2930  013520  037413  177776                BIT     @-2(R4),(R3)    ; CHECK IT
2931  013524  004737  017336                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
2932  013530  012746  125252                MOV     #125252,-(SP)   ; SET EVERY OTHER BIT ON THE STACK
2933  013534  017423  177776                MOV     @-2(R4),(R3)+   ; SET ALL THE BITS AT LOCATION TEMP
2934  013540  046643  000000                BIC     0(SP),-(R3)     ; CLEAR EVERY OTHER BIT AT LOCATION TEMP
2935  013544  022327  052525                CMP     (R3)+,#52525    ; TEMP SHOULD CONTAIN # 52525
2936  013550  001404                        BEQ     4$
```

```
2937  013552  012745  000302                MOV     #302,-(R5)
2938  013556  005245                        INC     -(R5)
2939  013560  000000                        HALT                    ; BIC FAILED IN MODE 6
2940  013562  012700  000446        4$:     MOV     #TEMP2+2,R0     ; PLACE THE ADDRESS OF TEMP2+2 IN R0
2941  013566  010340                        MOV     R3,-(R0)        ; PLACE THE ADDRESS OF TEMP1 IN TEMP2
2942  013570  014330                        MOV     -(R3),@(R0)+    ; MOVE # 52525 IN LOCATION TEMP1
2943  013572  000263                        SEVC                    ; SET V & C BITS
2944  013574  035026                        BIT     @-(R0),(SP)+    ; BIT TEST TEMP1 WITH STACK AND RESTORE STACK POINTER
2945  013576  004737  017252                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
2946  013602  020627  000530                CMP     SP,#START       ; MAKE SURE THAT THE SP IS OK
2947  013606  001404                        BEQ     INC1
2948  013610                        EBIC1:
2949  013610  012745  000303                MOV     #303,-(R5)
2950  013614  005245                        INC     -(R5)
2951  013616  000000                        HALT                    ; STACK POINTER FOULED UP OR SEQUENCE ERROR
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 102     NEW INSTRUCTIONS USED IN THIS SECTION ARE INC, DEC
                                        ;;*******************************************************************************

013620                                  INC1:
013620  021527  000102                          CMP     (R5),#102
013624  001404                                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
013626  012745  000304                          MOV     #304,-(R5)
013632  005245                                  INC     -(R5)
013634  000000                                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
013636  005215                          2$:     INC     (R5)
013640  012704  000442                          MOV     #TEMP1,R4       ; LOAD ADDRESS
013644  012714  077777                          MOV     #77777,(R4)     ; TEMP1 = 77777
013650  000261                                  SEC
013652  005214                                  INC     (R4)            ; ADD ONES INTO LOCATION
013654  004737  017400                          JSR     PC,@#$CC13      ; CHECK FOR CC = 13
013660  012714  177776                          MOV     #177776,(R4)
013664  012700  000440                          MOV     #TEMP,R0        ; R0 IS POINTING TO LOCATION TEMP
013670  012710  017336                          MOV     #$CC11,(R0)     ; PLACE THE ADDRESS OF SUBROUTINE TO CHECK CC = 11
                                                                        ; IN LOCATION TEMP
013674  005214                                  INC     (R4)            ;
013676  004730                                  JSR     PC,@(R0)+       ; CHECK FOR CC = 11
013700  005214                                  INC     (R4)            ;
013702  004737  017252                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
013706  005214                                  INC     (R4)            ;
013710  004737  017146                          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
013714  026427  000000  000001                  CMP     0(R4),#1        ; CHECK IT
013722  001404                                  BEQ     4$              ; CONTINUE IF OK
013724  012745  000305                          MOV     #305,-(R5)
013730  005245                                  INC     -(R5)
013732  000000                                  HALT                    ; INC INSTRUCTION FAILED
013734  000261                          4$:     SEC
013736  005314                                  DEC     (R4)            ; SUBTRACT ONES FROM LOCATION
013740  004737  017252                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
013744  005314                                  DEC     (R4)            ;
013746  004770  177776                          JSR     PC,@-2(R0)      ; CHECK FOR CC = 11
013752  012714  100000                          MOV     #100000,(R4)
013756  005314                                  DEC     (R4)            ;
013760  004737  017206                          JSR     PC,@#$CC3       ; CHECK FOR CC = 3
013764  005314                                  DEC     (R4)            ;
013766  004737  017146                          JSR     PC,@#$CC1       ; CHECK FOR CC = 1


                                        ;;*******************************************************************************
                                        ;*TEST: 103     NEW INSTRUCTION IN THIS SECTION IS COM
                                        ;;*******************************************************************************

013772                                  COM1:
013772  021527  000103                          CMP     (R5),#103
013776  001404                                  BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
014000  012745  000306                          MOV     #306,-(R5)
014004  005245                                  INC     -(R5)
014006  000000                                  HALT                    ; TEST IS IN WRONG SEQUENCE
014010  005215                          1$:     INC     (R5)
014012  012703  000442                          MOV     #TEMP1,R3       ; LOAD ADDRESS
```

T103 NEW INSTRUCTION IN THIS SECTION IS COM SEQ 0081

```
3008  014016  012713  125252          MOV     #125252,(R3)    ; LOAD EVERY OTHER BIT
3009  014022  000277                  SCC
3010  014024  005163  000000          COM     0(R3)           ; 1'S COMPLEMENT
3011  014030  004737  017146          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
3012  014034  022713  052525          CMP     #52525,(R3)     ; CHECK IT
3013  014040  001404                  BEQ     2$              ; CONTINUE IF OK
3014  014042  012745  000307          MOV     #307,-(R5)
3015  014046  005245                  INC     -(R5)
3016  014050  000000                  HALT                    ; COM INSTRUCTION FAILED
3017  014052  000277          2$:     SCC
3018  014054  005123                  COM     (R3)+           ; COMPLEMENT BACK
3019  014056  004737  017336          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
3020  014062  022743  125252          CMP     #125252,-(R3)   ; CHECK IT
3021  014066  001404                  BEQ     3$              ; CONTINUE IF OK
3022  014070  012745  000310          MOV     #310,-(R5)
3023  014074  005245                  INC     -(R5)
3024  014076  000000                  HALT                    ; COM INSTRUCTION FAILED
3025  014100  010300          3$:     MOV     R3,R0           ; R0 IS NOW POINTING TO LOCATION TEMP1
3026  014102  012710  177777          MOV     #177777,(R0)
3027  014106  000277                  SCC
3028  014110  005110                  COM     (R0)
3029  014112  004737  017252          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

```
3030                                    ;;*************************************************************************
3031                                    ;*TEST: 104     NEW INSTRUCTION IN THIS SECTION IS NEG
3032                                    ;;*************************************************************************
3033
3034  014116                            NEG1:
3035  014116  021527  000104                   CMP     (P5),#104
3036  014122  001033                           BNE     ENEG1                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
3037  014124  005215                    1$:    INC     (R5)
3038  014126  012704  000442                   MOV     #TEMP1,R4               ; LOAD ADDRESS
3039  014132  012724  000001                   MOV     #1,(R4)+                        ; LOAD THE LOCATION
3040  014136  010402                           MOV     R4,R2
3041  014140  012762  100000  000000           MOV     #100000,0(R2)
3042  014146  005444                           NEG     -(R4)                   ; 2'S COMPLEMENT
3043  014150  004737  017336                   JSR     PC,@#$CC11              ; CHECK FOR CC = 11
3044  014154  022724  177777                   CMP     #177777,(R4)+           ; CHECK IT
3045  014160  001404                           BEQ     2$                      ; CONTINUE IF OK
3046  014162  012745  000311                   MOV     #311,-(R5)
3047  014166  005245                           INC     -(R5)
3048  014170  000000                           HALT                            ; NEG INSTRUCTION FAILED
3049  014172  016444  000000           2$:    MOV     0(R4),-(R4)             ; TEMP1 CONTAINS THE LARGEST NEGATIVE NUMBER
3050  014176  005414                           NEG     (R4)                    ; 2'S COMPLEMENT
3051  014200  004737  017400                   JSR     PC,@#$CC13              ; CHECK FOR CC = 13
3052  014204  026214  000000                   CMP     0(R2),(R4)              ; CHECK IT
3053  014210  001404                           BEQ     ROL1                    ; CONTINUE IF OK
3054  014212                            ENEG1:
3055  014212  012745  000312                   MOV     #312,-(R5)
3056  014216  005245                           INC     -(R5)
3057  014220  000000                           HALT                            ; WRONG RESULT IN TEMP2 OR WRONG SEQUENCE
3058
3059
3060
3061                                    ;;*************************************************************************
3062                                    ;*TEST: 105     NEW INSTRUCTION IN THIS SECTION IS ROL
3063                                    ;;*************************************************************************
3064
3065  014222                            ROL1:
3066  014222  021527  000105                   CMP     (R5),#105
3067  014226  001032                           BNE     EROL1                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
3068  014230  005215                           INC     (R5)
3069  014232  012701  000444                   MOV     #TEMP2,R1               ; LOAD ADDRESS
3070  014236  012711  020000                   MOV     #20000,(R1)             ; LOAD LOCATION
3071  014242  000257                           CCC                             ; CLEAR FLAGS
3072  014244  006121                           ROL     (R1)+                   ; SHIFT
3073  014246  006141                           ROL     -(R1)                   ;
3074  014250  004737  017360                   JSR     PC,@#$CC12              ; CHECK FOR CC = 12
3075  014254  022711  100000                   CMP     #100000,(R1)            ; CHECK IT
3076  014260  001404                           BEQ     1$                      ; CONTINUE IF OK
3077  014262  012745  000313                   MOV     #313,-(R5)
3078  014266  005245                           INC     -(R5)
3079  014270  000000                           HALT                            ; ROL INSTRUCTION FAILED
3080  014272  006161  000000           1$:    ROL     0(R1)                   ; SHIFT
3081  014276  004737  017274                   JSR     PC,@#$CC7               ; CHECK FOR CC = 7
3082  014302  010102                           MOV     R1,R2                   ; R2 IS NOW POINTING TO LOCATION TEMP2
3083  014304  006112                           ROL     (R2)                    ; SHIFT
3084  014306  022711  000001                   CMP     #1,(R1)                 ; CHECK IT
3085  014312  001404                           BEQ     ROR1                    ; CONTINUE IF OK
```

```
  3086  014314                          EROL1:
  3087  014314  012745  000314                  MOV     #314,-(R5)
  3088  014320  005245                          INC     -(R5)
  3089  014322  000000                          HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE
```

```
3090
3091                                    ;;*******************************************************************************
3092                                    ;*TEST: 106     NEW INSTRUCTION IN THIS SECTION IS ROR
3093                                    ;;*******************************************************************************
3094  014324
3095  014324  021527  000106           ROR1:   CMP     (P5),#106
3096  014330  001030                           BNE     EROR1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
3097  014332  005215                           INC     (R5)
3098  014334  012702  000444                   MOV     #TEMP2,R2       ; LOAD ADDRESS
3099  014340  012712  000004                   MOV     #4,(R2)         ; LOAD LOCATION
3100  014344  000257                           CCC                     ; CLEAR FLAGS
3101  014346  006012                           ROR     (R2)            ; SHIFT
3102  014350  006012                           ROR     (R2)            ;
3103  014352  022712  000001                   CMP     #1,(R2)         ; CHECK IT
3104  014356  001404                           BEQ     1$              ; CONTINUE IF OK
3105  014360  012745  000315                   MOV     #315,-(R5)
3106  014364  005245                           INC     -(R5)
3107  014366  000000                           HALT                    ; ROR INSTRUCTION FAILED
3108  014370  006012           1$:     ROR     (R2)            ; SHIFT
3109  014372  004737  017274                   JSR     PC,@#$CC7       ; CHECK FOR CC = 7
3110  014376  006012                           ROR     (R2)            ; SHIFT
3111  014400  004737  017360                   JSR     PC,@#$CC12      ; CHECK FOR CC = 12
3112  014404  022712  100000                   CMP     #100000,(R2)    ; CHECK IT
3113  014410  001404                           BEQ     ASL1            ; CONTINUE IF OK
3114  014412                           EROR1:
3115  014412  012745  000316                   MOV     #316,-(R5)
3116  014416  005245                           INC     -(R5)
3117  014420  000000                           HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE
3118
3119
3120
3121                                    ;;*******************************************************************************
3122                                    ;*TEST: 107     NEW INSTRUCTION IN THIS SECTION IS ASL
3123                                    ;;*******************************************************************************
3124
3125  014422                           ASL1:
3126  014422  021527  000107                   CMP     (R5),#107
3127  014426  001404                           BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
3128  014430  012745  000317                   MOV     #317,-(R5)
3129  014434  005245                           INC     -(R5)
3130  014436  000000                           HALT                    ; PROGRAM IS IN WRONG SEQUENCE
3131  014440  005215           2$:     INC     (R5)
3132  014442  012703  000444                   MOV     #TEMP2,R3       ; LOAD ADDRESS
3133  014446  012713  020000                   MOV     #20000,(R3)     ; LOAD LOCATION
3134  014452  000257                           CCC                     ; CLEAR FLAGS
3135  014454  006313                           ASL     (R3)            ; SHIFT
3136  014456  006313                           ASL     (R3)            ;
3137  014460  004737  017360                   JSR     PC,@#$CC12      ; CHECK FOR CC = 12
3138  014464  022713  100000                   CMP     #100000,(R3)    ; CHECK IT
3139  014470  001404                           BEQ     4$              ; CONTINUE IF OK
3140  014472  012745  000320                   MOV     #320,-(R5)
3141  014476  005245                           INC     -(R5)
3142  014500  000000                           HALT                    ; ASL INSTRUCTION FAILED
3143  014502  006313           4$:     ASL     (R3)            ; SHIFT
3144  014504  004737  017274                   JSR     PC,@#$CC7       ; CHECK FOR CC = 7
3145  014510  006313                           ASL     (R3)            ; SHIFT
```

```
  3146  014512  004737  017230                JSR     PC,@#$CC4       : CHECK FOR CC = 4
```

```
3147
3148                                    ;;*******************************************************************************
3149                                    ;*TEST: 110     NEW INSTRUCTION IN THIS SECTION IS ASR
3150                                    ;;*******************************************************************************
3151   014516
3152   014516  021527  000110          ASR1:   CMP     (PS),#110
3153   014522  001040                          BNE     EASR1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
3154   014524  005215                  1$:     INC     (R5)
3155   014526  012704  000444                  MOV     #TEMP2,R4       ; LOAD ADDRESSES
3156   014532  012703  000440                  MOV     #TEMP,R3        ;
3157   014536  012714  000004                  MOV     #4,(R4)         ; LOAD LOCATION
3158   014542  000257                          CCC                     ; CLEAR FLAGS
3159   014544  006214                          ASR     (R4)            ; SHIFT
3160   014546  006214                          ASR     (R4)            ;
3161   014550  022714  000001                  CMP     #1,(R4)         ; CHECK IT
3162   014554  001404                          BEQ     2$              ; CONTINUE IF OK
3163   014556  012745  000321                  MOV     #321,-(R5)
3164   014562  005245                          INC     -(R5)
3165   014564  000000                          HALT                    ; ASR INSTRUCTION FAILED
3166   014566  006214                  2$:     ASR     (R4)            ; SHIFT
3167   014570  004737  017274                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
3168   014574  006214                          ASR     (R4)            ; SHIFT
3169   014576  004737  017230                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
3170   014602  012713  100002                  MOV     #100002,(R3)    ; LOAD LOCATION
3171   014606  006213                          ASR     (R3)            ; SHIFT
3172   014610  006213                          ASR     (R3)            ;
3173   014612  004737  017336                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
3174   014616  022713  160000                  CMP     #160000,(R3)    ; CHECK IT
3175   014622  001404                          BEQ     ADC1            ; CONTINUE IF OK
3176   014624                          EASR1:
3177   014624  012745  000322                  MOV     #322,-(R5)
3178   014630  005245                          INC     -(R5)
3179   014632  000000                          HALT                    ; WRONG RESULT IN TEMP OR WRONG SEQUENCE
3180
3181
3182
3183                                    ;;*******************************************************************************
3184                                    ;*TEST: 111     NEW INSTRUCTION IN THIS SECTION IS ADC
3185                                    ;;*******************************************************************************
3186
3187   014634                          ADC1:
3188   014634  021527  000111                  CMP     (R5),#111
3189   014640  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
3190   014642  012745  000323                  MOV     #323,-(R5)
3191   014646  005245                          INC     -(R5)
3192   014650  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
3193   014652  005215                  2$:     INC     (R5)
3194   014654  012700  000440                  MOV     #TEMP,R0        ; LOAD ADDRESS
3195   014660  005010                          CLR     (R0)            ; CLEAR THE LOCATION
3196   014662  000257                          CCC                     ; CLEAR FLAGS
3197   014664  005510                          ADC     (R0)            ; ADD C BIT = 0
3198   014666  004737  017230                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
3199   014672  000261                          SEC                     ; C=1
3200   014674  005510                          ADC     (R0)            ; ADD C BIT=1
3201   014676  000261                          SEC                     ; C=1
3202   014700  005510                          ADC     (R0)            ; AGAIN
```

```
3203  014702  004737  017126          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
3204  014706  022710  000002          CMP     #2,(R0)         ; CHECK IT
3205  014712  001404                  BEQ     4$              ; CONTINUE IF OK
3206  014714  012745  000324          MOV     #324,-(R5)
3207  014720  005245                  INC     -(R5)
3208  014722  000000                  HALT                    ; ADC INSTRUCTION FAILED
3209  014724  012710  077777  4$:     MOV     #77777,(R0)     ; LOAD LARGEST POSITVE NUMBER
3210  014730  000261                  SEC                     ; C=1
3211  014732  005510                  ADC     (R0)            ; ADD C BIT=1
3212  014734  004737  017360          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
3213  014740  022710  100000          CMP     #100000,(R0)    ; CHECK IT
3214  014744  001404                  BEQ     6$              ; CONTINUE IF OK
3215  014746  012745  000325          MOV     #325,-(R5)
3216  014752  005245                  INC     -(R5)
3217  014754  000000                  HALT                    ; ADC INSTRUCTION AILED
3218  014756  012710  177777  6$:     MOV     #-1,(R0)        ; LOAD -1
3219  014762  000261                  SEC                     ; C=1
3220  014764  005510                  ADC     (R0)            ; ADD C BIT=1
3221  014766  004737  017252          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

```
3222
3223                    ;;*******************************************************************************
3224                    ;*TEST: 112     NEW INSTRUCTION IN THIS SECTION IS SBC
3225                    ;;*******************************************************************************
3226  014772
3227  014772  021527  000112   SBC1:   CMP     (PS),#112
3228  014776  001404                   BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
3229  015000  012745  000326           MOV     #326,-(R5)
3230  015004  005245                   INC     -(R5)
3231  015006  000000                   HALT                            ; TEST IS IN WRONG SEQUENCE
3232  015010  005215           1$:     INC     (R5)
3233  015012  012701  000440           MOV     #TEMP,R1                ; LOAD ADDRESS
3234  015016  012711  000003           MOV     #3,(R1)                 ; LOAD LOCATION
3235  015022  000257                   CCC                             ; CLEAR FLAGS
3236  015024  005611                   SBC     (R1)                    ; SUBTRACT C BIT=0
3237  015026  004737  017126           JSR     PC,@#$CC0               ; CHECK FOR CC = 0
3238  015032  022711  000003           CMP     #3,(R1)                 ; CHECK IT
3239  015036  001404                   BEQ     2$                      ; CONTINUE IF OK
3240  015040  012745  000327           MOV     #327,-(R5)
3241  015044  005245                   INC     -(R5)
3242  015046  000000                   HALT                            ; SBC INSTRUCTION FAILED
3243  015050  000261           2$:     SEC                             ; C=1
3244  015052  005611                   SBC     (R1)                    ; SUBTRACT C BIT=1
3245  015054  000261                   SEC                             ; C=1
3246  015056  005611                   SBC     (R1)                    ;
3247  015060  004737  017126           JSR     PC,@#$CC0               ; CHECK FOR CC = 0
3248  015064  022711  000001           CMP     #1,(R1)                 ; CHECK IT
3249  015070  001404                   BEQ     3$
3250  015072  012745  000330           MOV     #330,-(R5)
3251  015076  005245                   INC     -(R5)
3252  015100  000000                   HALT                            ; SBC INSTRUCTION FAILED
3253  015102  000261           3$:     SEC                             ; C=1
3254  015104  005611                   SBC     (R1)                    ; SUBTRACT C BIT=1
3255  015106  004737  017230           JSR     PC,@#$CC4               ; CHECK FOR CC = 4
3256  015112  000261                   SEC                             ; C=1
3257  015114  005611                   SBC     (R1)                    ; SUBTRACT C BIT = 1
3258  015116  004737  017336           JSR     PC,@#$CC11              ; CHECK FOR CC = 11
3259  015122  022711  177777           CMP     #-1,(R1)                ; CHECK IT
3260  015126  001404                   BEQ     4$                      ; CONTINUE IF OK
3261  015130  012745  000331           MOV     #331,-(R5)
3262  015134  005245                   INC     -(R5)
3263  015136  000000                   HALT                            ; SBC INSTRUCTION FAILED
3264  015140  012711  100000   4$:     MOV     #100000,(R1)            ; LOAD R1
3265  015144  000261                   SEC                             ; C=1
3266  015146  005611                   SBC     (R1)                    ; SUBTRACT C BIT = 1
3267  015150  004737  017166           JSR     PC,@#$CC2               ; CHECK FOR CC = 2
3268
3269
3270
3271                    ;;*******************************************************************************
3272                    ;*TEST: 113     NEW INSTRUCTION IN THIS SECTION IS SXT
3273                    ;;*******************************************************************************
3274
3275  015154   SXT1:
3276  015154  021527  000113           CMP     (R5),#113
3277  015160  001026                   BNE     ESXT1                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
```

```
3278  015162  005215                  1$:     INC     (R5)
3279  015164  012702  000442                  MOV     #TEMP1,R2       ; LOAD ADDRESS
3280  015170  005012                          CLR     (R2)            ; CLEAR LOCATIONS
3281  015172  000277                          SCC
3282  015174  000254                          CLNZ
3283  015176  006712                          SXT     (R2)            ; SIGN EXTEND
3284  015200  004737  017252                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
3285  015204  005712                          TST     (R2)            ; LOCATION SHOULD STILL BE 0
3286  015206  001404                          BEQ     2$              ; CONTINUE IF OK
3287  015210  012745  000332                  MOV     #332,-(R5)
3288  015214  005245                          INC     -(R5)
3289  015216  000000                          HALT                    ; SXT INSTRUCTION FAILED
3290  015220  000273                  2$:     SENVC                   ; SET N, V & C BITS
3291  015222  006712                          SXT     (R2)            ; SIGN EXTEND
3292  015224  004737  017336                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
3293  015230  022712  177777                  CMP     #-1,(R2)        ; LOCATION SHOULD NEW HAVE -1
3294  015234  001404                          BEQ     SWAB1           ; CONTINUE IF OK
3295  015236                          ESXT1:
3296  015236  012745  000333                  MOV     #333,-(R5)
3297  015242  005245                          INC     -(R5)
3298  015244  000000                          HALT                    ; WRONG RESULT IN TEMP1 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;*TEST: 114     NEW INSTRUCTIOON IN THIS SECTION IS SWAB
                                ;;*******************************************************************************

                                SWAB1:
015246                                  CMP     (P5),#114
015246  021527  000114                  BNE     ESWAB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
015252  001034                          INC     (R5)
015254  005215                          MOV     #TEMP2,R3       ; LOAD ADDRESS
015256  012703  000444                  MOV     #125125,(R3)    ; LOAD BIT PATTERN INTO LOCATION
015262  012713  125125                  SCC
015266  000277                          CLN
015270  000250                          SWAB    (R3)            ; SWAP BYTES OF LOCATIONS
015272  000313                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
015274  004737  017316                  CMP     #52652,(R3)     ; CHECK IT
015300  022713  052652                  BEQ     1$              ; CONTINUE IF OK
015304  001404                          MOV     #334,-(R5)
015306  012745  000334                  INC     -(R5)
015312  005245                          HALT                    ; SWAB INSTRUCTION FAILED
015314  000000                  1$:     MOV     #377,(R3)
015316  012713  000377                  SCC
015322  000277                          CLZ
015324  000244                          SWAB    0(R3)
015326  000363  000000                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
015332  004737  017230                  CMP     #177400,(R3)
015336  022713  177400                  BEQ     XOR1
015342  001404                  ESWAB1:
015344                                  MOV     #335,-(R5)
015344  012745  000335                  INC     -(R5)
015350  005245                          HALT                    ; WRONG RESULT IN: TEMP2 OR WRONG SEQUENCE
015352  000000
```

```
                                ;;*******************************************************************************
                                ;*TEST: 115     NEW INSTRUCTION IN THIS SECTION IS XOR
                                ;;*******************************************************************************

015354                          XOR1:
015354  021527  000115                  CMP     (R5),#115
015360  001041                          BNE     EXOR1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
015362  005215                          INC     (R5)
015364  012704  177777                  MOV     #-1,R4          ; LOAD LOCATIONS
015370  012767  177777  163044          MOV     #-1,TEMP1       ;
015376  000277                          SCC
015400  074467  163036                  XOR     R4,TEMP1        ; SHOULD PRODUCE 0'S IN TEMP1
015404  004737  017252                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
015410  012767  077777  163024          MOV     #77777,TEMP1
015416  012700  000442                  MOV     #TEMP1,R0       ; PLACE THE ADDRESS OF TEMP1 IN R0
015422  000263                          SEVC                    ; SET V & C BITS
015424  000244                          CLZ
015426  074410                          XOR     R4,(R0)
015430  004737  017336                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
015434  012701  125252                  MOV     #125252,R1      ; LOAD LOCATIONS
015440  012720  052525                  MOV     #52525,(R0)+    ;
015444  000277                          SCC
015446  074140                          XOR     R1,-(R0)        ; SHOULD PRODUCE ALL 1'S IN TEMP1
```

```
DVKAAB  MACY11 30(1046)  25-OCT-77  13:11  PAGE 91
DVKAAB.P11    25-OCT-77 13:09          T115    NEW INSTRUCTION IN THIS SECTION IS XOR                                  SEQ 0091

  3355  015450  004737  017336                      JSR     PC,@#$CC11      ; CHECK FOR CC = 11
  3356  015454  022737  177777  000442              CMP     #-1,@#TEMP1     ; CHECK IT
  3357  015462  001404                              BEQ     ADD1            ; CONTINUE IF OK
  3358  015464                              EXOR1:
  3359  015464  012745  000336                      MOV     #336,-(R5)
  3360  015470  005245                              INC     -(R5)
  3361  015472  000000                              HALT                    ; WRONG RESULT IN TEMP1 OR WRONG SEQUENCE
```

```
3362                                    ;;*****************************************************************************
3363                                    ;*TEST: 116     NEW INSTRUCTION IN THIS SECTION IS ADD
3364                                    ;;*****************************************************************************
3365
3366  015474                            ADD1:
3367  015474  021527  000116                   CMP     (P5),#116
3368  015500  001133                           BNE     EADD1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
3369  015502  005215                           INC     (R5)
3370  015504  012700  000444                   MOV     #TEMP2,R0       ; LOAD ADDRESSES
3371  015510  012701  000440                   MOV     #TEMP,R1        ;
3372  015514  012767  021421  162722           MOV     #21421,TEMP2    ; LOAD LOCATIONS
3373  015522  011011                           MOV     (R0),(R1)       ;
3374  015524  061011                           ADD     (R0),(R1)       ; ADD
3375  015526  004737  017126                   JSR     PC,@#$CC0       ; CHECK FOR CC = 0
3376  015532  022767  043042  162700           CMP     #43042,TEMP     ; CHECK IT
3377  015540  001404                           BEQ     1$              ; CONTINUE IF OK
3378  015542  012745  000337                   MOV     #337,-(R5)
3379  015546  005245                           INC     -(R5)
3380  015550  000000                           HALT                    ; ADD INSTRUCTION FAILED
3381  015552  005010                   1$:     CLR     (R0)            ; CLEAR LOCATION TEMP2
3382  015554  060020                           ADD     R0,(R0)+        ; PLACE THE ADDRESS OF TEMP2 IN LOCATION TEMP2
3383  015556  024027  000444                   CMP     -(R0),#TEMP2    ; CHECK IT
3384  015562  001404                           BEQ     2$
3385  015564  012745  000340                   MOV     #340,-(R5)
3386  015570  005245                           INC     -(R5)
3387  015572  000000                           HALT                    ; ADD INSTRUCTION FAILED IN MODE 2
3388  015574  012767  156357  162642   2$:     MOV     #-21421,TEMP2   ; LOAD LOCATIONS
3389  015602  012011                           MOV     (R0)+,(R1)      ;
3390  015604  064011                           ADD     -(R0),(R1)      ; ADD
3391  015606  004737  017336                   JSR     PC,@#$CC11      ; CHECK FOR CC = 11
3392  015612  022767  134736  162620           CMP     #-43042,TEMP    ; CHECK IT
3393  015620  001404                           BEQ     3$              ;CONTINUE IF OK
3394  015622  012745  000341                   MOV     #341,-(R5)
3395  015626  005245                           INC     -(R5)
3396  015630  000000                           HALT                    ; ADD INSTRUCTION FAILED
3397  015632  012767  100000  162604   3$:     MOV     #100000,TEMP2   ; LOAD LOCATIONS
3398  015640  011061  000000                   MOV     (R0),0(R1)      ;
3399  015644  066011  000000                   ADD     0(R0),(R1)      ; ADD SHOULD RESULT AS 0'S
3400  015650  004737  017274                   JSR     PC,@#$CC7       ; CHECK FOR CC=7
3401  015654  012767  021421  162560           MOV     #21421,TEMP1    ; LOAD LOCATION TEMP1
3402  015662  012760  000442  000000           MOV     #TEMP1,0(R0)    ; PLACE THE ADDRESS OF LOCATION TEMP1 IN TEMP2
3403  015670  012711  156357                   MOV     #-21421,(R1)    ; LOAD LOCATION TEMP
3404  015674  010004                           MOV     R0,R4           ; MAKE R4 POINT TO LOCATION TEMP2
3405  015676  067411  000000                   ADD     @0(R4),(R1)     ; ADD SHOULD RESULT AS 0'S
3406  015702  004737  017252                   JSR     PC,@#$CC5       ; CHECK FOR CC=5
3407  015706  005430                           NEG     @(R0)+          ; NEGATE THE CONTENTS OF TEMP1
3408  015710  012746  021421                   MOV     #21421,-(SP)    ; PLACE # 21421 ON THE STACK
3409  015714  065066  000000                   ADD     @-(R0),0(SP)    ; ADD, SHOULD=0'S
3410  015720  004737  017252                   JSR     PC,@#$CC5       ; CHECK FOR CC=5
3411  015724  005726                           TST     (SP)+           ; CHECK THE STACK TO CONTAIN 0, ALSO
3412                                                                   ; RESTORE THE STACK POINTER
3413  015726  001404                           BEQ     4$
3414  015730  012745  000342                   MOV     #342,-(R5)
3415  015734  005245                           INC     -(R5)
3416  015736  000000                           HALT                    ; ADD INSTRUCTION FAILED IN MODE 5
3417  015740  012767  137777  162476   4$:     MOV     #137777,TEMP2
```

```
3418  015746  062767  137777  162470          ADD     #137777,TEMP2
3419  015754  004737  017206                  JSR     PC,@#$CC3       ; CHECK CC=3
3420  015760  022767  077776  162456          CMP     #77776,TEMP2
3421  015766  001404                          BEQ     SUB1
3422  015770                          EADD1:
3423  015770  012745  000343                  MOV     #343,-(R5)
3424  015774  005245                          INC     -(R5)
3425  015776  000000                          HALT                    ; WRONG RESULT AT TEMP OR WRONG SEQUENCE
3426
3427
3428
3429                                  ;;**********************************************************************
3430                                  ;*TEST: 117     NEW INSTRUCTION IN THIS SECTION IS SUB
3431                                  ;;**********************************************************************
3432
3433  016000                          SUB1:
3434  016000  021527  000117                  CMP     (R5),#117
3435  016004  001100                          BNE     ESUB1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
3436  016006  005215                          INC     (R5)
3437  016010  012702  000440                  MOV     #TEMP,R2        ; LOAD ADDRESSES
3438  016014  012703  000442                  MOV     #TEMP1,R3       ;
3439  016020  012767  021421  162412          MOV     #21421,TEMP     ; LOAD LOCATIONS
3440  016026  012767  156357  162406          MOV     #-21421,TEMP1   ;
3441  016034  161213                          SUB     (R2),(R3)       ; RESULT SHOULD=-43042
3442  016036  004737  017316                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
3443  016042  022767  134736  162372          CMP     #-43042,TEMP1   ; CHECK IT
3444  016050  001404                          BEQ     1$              ;CONTINUE IF OK
3445  016052  012745  000344                  MOV     #344,-(R5)
3446  016056  005245                          INC     -(R5)
3447  016060  000000                          HALT                    ; SUB INSTRUCTION FAILED
3448  016062  012767  021421  162352  1$:     MOV     #21421,TEMP1    ; LOAD LOCATION
3449  016070  161213                          SUB     (R2),(R3)       ; RESULT SHOULD=0
3450  016072  001404                          BEQ     2$
3451  016074  012745  000345                  MOV     #345,-(R5)
3452  016100  005245                          INC     -(R5)
3453  016102  000000                          HALT                    ; SUB INSTRUCTION FAILED
3454  016104  012767  177777  162330  2$:     MOV     #-1,TEMP1       ; LOAD LOCATIONS
3455  016112  012767  077777  162320          MOV     #77777,TEMP     ; LOAD LOCATIONS
3456  016120  161312                          SUB     (R3),(R2)       ; RESULT SHOULD GIVE 100000 AND OVERFLOW
3457  016122  004737  017400                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
3458  016126  022767  100000  162304          CMP     #100000,TEMP    ; CHECK IT
3459  016134  001404                          BEQ     3$              ; CONTINUE IF OK
3460  016136  012745  000346                  MOV     #346,-(R5)
3461  016142  005245                          INC     -(R5)
3462  016144  000000                          HALT                    ; SUB INSTRUCTION FAILED
3463  016146  012712  177777          3$:     MOV     #-1,(R2)
3464  016152  161312                          SUB     (R3),(R2)
3465  016154  004737  017230                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
3466  016160  012767  077777  162252          MOV     #77777,TEMP
3467  016166  162767  077777  162244          SUB     #77777,TEMP
3468  016174  004737  017230                  JSR     PC,@#$CC4       ; CHECK FOR CC=4
3469  016200  005767  162234                  TST     TEMP
3470  016204  001404                          BEQ     SOB             ; TEMP SHOULD BE =0
3471  016206                          ESUB1:
3472  016206  012745  000347                  MOV     #347,-(R5)
3473  016212  005245                          INC     -(R5)
```

```
  3474  016214  000000                    HALT                    ; SUB INSTRUCTION FAILED OR SEQUENCE ERROR
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 120     NEW INSTRUCTION IN THIS SECTION IS SOB
                                        ;;*******************************************************************************

016216                                  SOB:
016216  021527  000120                          CMP     (P5),#120
016222  001042                                 BNE     ESOB            ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
016224  005215                                 INC     (R5)
016226  012700  000012                         MOV     #10.,R0         ; LOAD REGISTERS
016232  005001                                 CLR     R1              ;
016234  005201                          1$:    INC     R1              ; KEEP COUNT
016236  020127  000012                         CMP     R1,#10.
016242  003404                                 BLE     2$
016244  012745  000350                         MOV     #350,-(R5)
016250  005245                                 INC     -(R5)
016252  000000                                 HALT                    ; SOB INSTRUCTION FAILED
016254  000277                          2$:    SCC
016256  077012                                 SOB     R0,1$           ; SUB. 1 FROM REG. 0, GO BACK TO 1$
016260  004737  017420                         JSR     PC,@#$CC17      ; CHECK FOR CC = 17
016264  005700                                 TST     R0              ; REG. 0 = 0 ?
016266  001404                                 BEQ     3$              ; NO, FAILED
016270  012745  000351                         MOV     #351,-(R5)
016274  005245                                 INC     -(R5)
016276  000000                                 HALT                    ; SOB INSTRUCTION FAILED
016300  022701  000012                  3$:    CMP     #10.,R1         ; DID IT GO THRU 10 TIMES ?
016304  001404                                 BEQ     4$              ; CONTINUE IF OK
016306  012745  000352                         MOV     #352,-(R5)
016312  005245                                 INC     -(R5)
016314  000000                                 HALT                    ; SOB INSTRUCTION FAILED
016316  012704  000010                  4$:    MOV     #10,R4          ; PLACE #10 IN R4
016322  077401                          5$:    SOB     R4,5$           ; STAY HERE UNTILL R4 = 0
016324  005704                                 TST     R4
016326  001404                                 BEQ     PSWNO           ; CONTINUE IF OK
016330                                  ESOB:
016330  012745  000353                         MOV     #353,-(R5)
016334  005245                                 INC     -(R5)
016336  000000                                 HALT                    ; SOB FAILED OR WRONG SEQUENCE
                                        ;;*******************************************************************************
                                        ;*TEST: 121     NEW INSTRUCTIONS IN THIS SECTION ARE MTPS & MFPS
                                        ;;*******************************************************************************

016340                                  PSWNO:
016340  021527  000121                         CMP     (R5),#121
016344  001042                                 BNE     EPSWNO          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
016346  005215                                 INC     (R5)
016350  012700  000440                         MOV     #TEMP,R0        ; PUT THE ADDRESS OF TEMP IN R0
016354  012701  000442                         MOV     #TEMP1,R1       ; PUT THE ADDRESS OF TEMP1 IN R1
016360  012711  177777                         MOV     #177777,(R1)    ; TEMP1 = 177777
016364  005010                                 CLR     (R0)            ; TEMP  = 0
016366                                         MTPS    (R0)            ; PSW = 0
016366  106410                                 .WORD   106400!..C
016370  004737  017126                         JSR     PC,@#$CC0       ; CHECK FOR CC = 0
```

T121 NEW INSTRUCTIONS IN THIS SECTION ARE MTPS & MFPS SEQ 0096

```
3531  016374                          MFPS    (R1)            ; MOVE PSW TO TEMP1
3532  016374  106711                  .WORD   106700!..C
3533  016376  004737  017230          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
3534  016402  022711  177400          CMP     #177400,(R1)    ; CHECK TEMP1 TO MAKE SURE THAT  ONLY
3535                                                          ; THE LOWER BYTE WAS AFFECTED BY MFPS
3536  016406  001404                  BEQ     1$
3537  016410  012745  000354          MOV     #354,-(R5)
3538  016414  005245                  INC     -(R5)
3539  016416  000000                  HALT                    ; MTPS OR MFPS INSTRUCTION FAILED
3540  016420  005011          1$:     CLR     (R1)
3541  016422                          MTPS    #377            ; SET PSW = 357 SINCE T BIT CAN NOT BE SET BY MTPS
3542  016422  106427                  .WORD   106400!..C
3543  016426  004737  017420          JSR     PC,@#$CC17      ; CHECK FOR CC = 17
3544  016432                          MFPS    TEMP1           ; MOVE PSW TO TEMP1
3545  016432  106767                  .WORD   106700!..C
3546  016436  004737  017336          JSR     PC,@#$CC11      ; CHECK FOR CC = 11 (C BIT SHOULD NOT BE EFFECTED BY MFP
3547  016442  022767  000357  161772  CMP     #357,TEMP1
3548  016450  001404                  BEQ     BTWRD
3549  016452                  EPSWNO:
3550  016452  012745  000355          MOV     #355,-(R5)
3551  016456  005245                  INC     -(R5)
3552  016460  000000                  HALT                    ; MFPS INSTRUCTION FAILED IN MODE 6
3553                                                          ; OR SEQUENCE ERROR
```

```
3554                                    ;;*************************************************************************
3555                                    ;*TEST: 122     BYTE INSTRUCTIONS REQUIRING WORD INST. TO CHECK
3556                                    ;;*************************************************************************
3557
3558  016462                            BTWRD:
3559  016462  021527  000122                   CMP     (P5),#122
3560  016466  001124                           BNE     EBTWRD          ; IF IN WRONG SEQUNCE GO TO HALT AT THE END OF THE TEST
3561  016470  005215                           INC     (R5)
3562  016472  005000                           CLR     R0
3563  016474  000277                           SCC
3564  016476  112700  000200                  MOVB    #200,R0         ; SET THE HIGHEST BIT OF THE
3565                                                                    ; LOWER BYTE
3566  016502  004737  017336                   JSR     PC,@#$CC11      ; CHECK FOR CC=11
3567  016506  022700  177600                   CMP     #177600,R0      ; CHECK FOR SIGN EXTENSION IN R0
3568  016512  001404                           BEQ     1$
3569  016514  012745  000356                   MOV     #356,-(R5)
3570  016520  005245                           INC     -(R5)
3571  016522  000000                           HALT                    ; SIGN WAS NOT EXTENDED IN R0
3572  016524  000277                   1$:     SCC
3573  016526  012700  177777                   MOV     #177777,R0
3574  016532  112700  000000                  MOVB    #0,R0           ;CLEAR THE LOWER BYTE OF R0.
3575  016536  004737  017252                   JSR     PC,@#$CC5       ; CHECK FOR CC=5
3576  016542  005700                           TST     R0              ; CHECK R0 FOR SIGN EXTENTION
3577  016544  001404                           BEQ     2$
3578  016546  012745  000357                   MOV     #357,-(R5)
3579  016552  005245                           INC     -(R5)
3580  016554  000000                           HALT                    ; SIGN WAS NOT EXTENDED IN R0.
3581  016556  012704  000444           2$:     MOV     #TEMP2,R4       ; R4 IS POINTING TO TEMP2
3582  016562  012714  000377                   MOV     #377,(R4)       ; PLACE #377 IN LOCATION TEMP2
3583  016566  012706  000526                   MOV     #START-2,R6
3584  016572  116426  000000                  MOVB    0(R4),(R6)+     ; PUSH # 377 ON STACK
3585  016576  022706  000530                   CMP     #START,R6
3586  016602  001404                           BEQ     3$
3587  016604  012745  000360                   MOV     #360,-(R5)
3588  016610  005245                           INC     -(R5)
3589  016612  000000                           HALT
3590                                                                    ; R6 DID NOT GET INCREMENTED
3591  016614  124627  000377           3$:     CMPB    -(R6),#377      ; BY 2 BY A BYTE INSTRUCTION
3592                                                                    ; CHECK LOCATION START-2 TO
3593  016620  001404                           BEQ     4$              ; CONTAIN PROPER DATA
3594  016622  012745  000361                   MOV     #361,-(R5)
3595  016626  005245                           INC     -(R5)
3596  016630  000000                           HALT                    ; BYTE INSTRUCTION IS FAILING WITH R6
3597  016632  022706  000526           4$:     CMP     #START-2,R6     ; CHECK THAT R6 WAS DECREMENTED
3598                                                                    ; BY 2 BY A BYTE INSTRUCTION
3599  016636  001404                           BEQ     5$
3600  016640  012745  000362                   MOV     #362,-(R5)
3601  016644  005245                           INC     -(R5)
3602  016646  000000                           HALT                    ; R6 WAS NOT DECREMENTED
3603  016650  016467  000000  161562   5$:     MOV     0(R4),TEMP      ; SET THE LOWER BYTE OF LOCATION TEMP
3604  016656  005726                           TST     (R6)+           ; RESTORE STACK POINTER
3605  016660  000277                           SCC
3606  016662  114667  161553                  MOVB    -(SP),TEMP+1    ; SET THE HIGHER BYTE OF LOCATION TEMP
3607  016666  004737  017336                   JSR     PC,@#$CC11      ; CHECK FOR CC=11
3608  016672  022767  177777  161540          CMP     #177777,TEMP    ; CHECK TEMP FOR THE CORRECT VALUE
3609  016700  001404                           BEQ     6$
```

```
 3610  016722  012745  000363                  MOV     #363,-(R5)
 3611  016726  005245                          INC     -(R5)
 3612  016730  000000                          HALT                    ; TEMP FOULED UP
 3613  016732  005067  161522          6$:     CLR     TEMP
 3614  016736  000241                          CLC
 3615  016740  105167  161515                  COMB    TEMP+1          ; WRITE 1'S IN THE HIGHER BYTE OF TEMP
 3616  016744  004737  017336                  JSR     PC,@#$CC11      ; CHECK FOR CC=11
 3617  016750  022767  177400  161502          CMP     #177400,TEMP
 3618  016756  001404                          BEQ     NEXT
 3619  016760                          EBTWRD:
 3620  016760  012745  000364                  MOV     #364,-(R5)
 3621  016764  005245                          INC     -(R5)
 3622  016766  000000                          HALT                    ; WRONG VALUE IN TEMP OR WRONG SEQUENCE
```

```
3623
3624
3625
3626                                    ;               END OF PASS
3627                                    ;               ***********
3628
3629
3630
3631   016750                           NEXT:
3632   016750   021527   000123                 CMP     (R5),#123
3633   016754   001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
3634   016756   012745   000365                 MOV     #365,-(R5)
3635   016762   005245                          INC     -(R5)
3636   016764   000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
3637   016766   005267   161414        2$:      INC     $PASS
3638   016772   126727   161410   000001       CMPB    $PASS,#1        ; ALLOW THE TYPE OUT OF END OF
3639                                                                    ; PASS EVERY 377 PASSES
3640   017000   001002                          BNE     GET42
3641   017002   000004   000454                 .TYPE   .ENDPAS         ; TYPE END OF PASS MESSAGE
3642   017006   013700   000042        GET42:   MOV     @#42,R0
3643   017012   001404                          BEQ     DOAGN
3644   017014   004710                $ENDAD:   JSR     PC,(R0)
3645   017016   000240                          NOP
3646   017020   000240                          NOP
3647   017022   000240                          NOP
3648   017024   005067   161354        DOAGN:   CLR     $TESTN          ; PREPARE TO START FROM TEST 0
3649   017030   000167   161474        RETURN:  JMP     START           ; START TEST OVER AT BEGINNING
3650                                    ;       ------------------------
3651
3652                                    ;;**************************************************************************
3653
3654                                    .SBTTL  POWER FAIL ROUTINE
3655
3656
3657   017034   012737   017044   000024 PWRDN:  MOV     #PWRUP,@#24     ; GO TO POWER UP ROUTINE AFTER THE POWER COMES BACK
3658   017042   000000                          HALT
3659
3660   017044   012706   000530        PWRUP:   MOV     #START,SP
3661   017050   012737   017034   000024       MOV     #PWRDN,@#24
3662   017056   000004   000470                 .TYPE   .POWER
3663   017062   000760                          BR      DOAGN
```

```
 3664
 3665
 3666                                   .SBTTL  TYPE ROUTINE
 3667
 3668
 3669  017064  132737  000040  000421  TYPE:   BITB    #40,@#$ENVM     ; HAS THE CONSOLE OUTPUTS BEEN SUPPRESSED?
 3670  017072  001012                   BNE     4$              ; IF SO THEN GO TO 4$
 3671  017074  017603  000000           MOV     @(SP),R3        ; GET ADDRESS OF MESSAGE
 3672
 3673  017100  105713                   1$:     TSTB    (R3)            ; END OF MESSAGE ?
 3674  017102  001406                   BEQ     4$              ; YES, GO WRAP IT UP
 3675
 3676  017104  105777  161336           3$:     TSTB    @TPS            ; READY FOR NEXT CHARACTER ?
 3677  017110  100375                   BPL     3$              ; NO, WAIT
 3678  017112  112377  161332           MOVB    (R3)+,@TPB      ; LOAD AND TYPE THE CHARACTER
 3679  017116  000770                   BR      1$              ; YES, GET THE NEXT CHARACTER
 3680
 3681  017120  062716  000002           4$:     ADD     #2,(SP)         ; ADJUST THE RETURN PC
 3682  017124  000006                   RTT                     ; RETURN
 3683
```

```
3684  017126  003402              $CC0:   BLE     1$
3685  017130  100401                      BMI     1$
3686  017132  103004                      BCC     2$
3687  017134                      1$:
3688  017134  012745  000366              MOV     #366,-(R5)
3689  017140  005245                      INC     -(R5)
3690  017142  000000                      HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 0
3691  017144  000207              2$:     RTS     PC
3692
3693  017146  003402              $CC1:   BLE     1$
3694  017150  100401                      BMI     1$
3695  017152  103404                      BCS     2$
3696  017154                      1$:
3697  017154  012745  000367              MOV     #367,-(R5)
3698  017160  005245                      INC     -(R5)
3699  017162  000000                      HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 1
3700  017164  000207              2$:     RTS     PC
3701
3702  017166  100402              $CC2:   BMI     1$
3703  017170  101401                      BLOS    1$
3704  017172  102404                      BVS     2$
3705  017174                      1$:
3706  017174  012745  000370              MOV     #370,-(R5)
3707  017200  005245                      INC     -(R5)
3708  017202  000000                      HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 2
3709  017204  000207              2$:     RTS     PC
3710
3711  017206  100403              $CC3:   BMI     1$
3712  017210  001402                      BEQ     1$
3713  017212  102001                      BVC     1$
3714  017214  103404                      BCS     2$
3715  017216                      1$:
3716  017216  012745  000371              MOV     #371,-(R5)
3717  017222  005245                      INC     -(R5)
3718  017224  000000                      HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 3
3719  017226  000207              2$:     RTS     PC
3720
3721  017230  100403              $CC4:   BMI     1$
3722  017232  001002                      BNE     1$
3723  017234  102401                      BVS     1$
3724  017236  103004                      BCC     2$
3725  017240                      1$:
3726  017240  012745  000372              MOV     #372,-(R5)
3727  017244  005245                      INC     -(R5)
3728  017246  000000                      HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 4
3729  017250  000207              2$:     RTS     PC
3730
3731  017252  100403              $CC5:   BMI     1$
3732  017254  001002                      BNE     1$
3733  017256  102401                      BVS     1$
3734  017260  103404                      BCS     2$
3735  017262                      1$:
3736  017262  012745  000373              MOV     #373,-(R5)
3737  017266  005245                      INC     -(R5)
3738  017270  000000                      HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 5
3739  017272  000207              2$:     RTS     PC
```

```
3740
3741
3742  017274  100403                  $CC7:   BMI     1$
3743  017276  001002                          BNE     1$
3744  017300  102001                          BVC     1$
3745  017302  103404                          BCS     2$
3746  017304                          1$:
3747  017304  012745  000374                  MOV     #374,-(R5)
3748  017310  005245                          INC     -(R5)
3749  017312  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 7
3750  017314  000207                  2$:     RTS     PC
3751
3752  017316  100002                  $CC10:  BPL     1$
3753  017320  101401                          BLOS    1$
3754  017322  102004                          BVC     2$
3755  017324                          1$:
3756  017324  012745  000375                  MOV     #375,-(R5)
3757  017330  005245                          INC     -(R5)
3758  017332  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 10
3759  017334  000207                  2$:     RTS     PC
3760
3761  017336  100003                  $CC11:  BPL     1$
3762  017340  001402                          BEQ     1$
3763  017342  102401                          BVS     1$
3764  017344  103404                          BCS     2$
3765  017346                          1$:
3766  017346  012745  000376                  MOV     #376,-(R5)
3767  017352  005245                          INC     -(R5)
3768  017354  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 11
3769  017356  000207                  2$:     RTS     PC
3770
3771  017360  100002                  $CC12:  BPL     1$
3772  017362  101401                          BLOS    1$
3773  017364  102404                          BVS     2$
3774  017366                          1$:
3775  017366  012745  000377                  MOV     #377,-(R5)
3776  017372  005245                          INC     -(R5)
3777  017374  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 12
3778  017376  000207                  2$:     RTS     PC
3779
3780  017400  100002                  $CC13:  BPL     1$
3781  017402  003401                          BLE     1$
3782  017404  103404                          BCS     2$
3783  017406                          1$:
3784  017406  012745  000400                  MOV     #400,-(R5)
3785  017412  005245                          INC     -(R5)
3786  017414  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 13
3787  017416  000207                  2$:     RTS     PC
3788
3789  017420  100003                  $CC17:  BPL     1$
3790  017422  001002                          BNE     1$
3791  017424  102001                          BVC     1$
3792  017426  103404                          BCS     2$
3793  017430                          1$:
3794  017430  012745  000401                  MOV     #401,-(R5)
3795  017434  005245                          INC     -(R5)
```

```
3796  017436  000000                        HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 17
3797  017440  000207              25:       RTS     PC
3798
3799          000001                        .END
```

```
ABASE = 000000            333
ACDW1 = 000000            333
ACDW2 = 000000            333
ACPUOP= 000000            333      348
ADCB0   004204           1287     1296#
ADCB1   012460           2696     2708#
ADC0    006000           1694     1703#
ADC1    014634           3175     3187#
ADDW0 = 000000            333
ADDW1 = 000000            333
ADDW10= 000000            333
ADDW11= 000000            333
ADDW12= 000000            333
ADDW13= 000000            333
ADDW14= 000000            333
ADDW15= 000000            333
ADDW2 = 000000            333
ADDW3 = 000000            333
ADDW4 = 000000            333
ADDW5 = 000000            333
ADDW6 = 000000            333
ADDW7 = 000000            333
ADDW8 = 000000            333
ADDW9 = 000000            333
ADD0    006604           1869     1881#
ADD1    015474           3357     3366#
ADEVCT= 000000            333      339
ADEVM = 000000            333
ADR     000430            380#     381     2098#   2101    2118    2213#   2216    2242    2307#   2310    2856    2897
ADR1    000432            382#     383     2099#   2102    2119    2214#   2217    2243    2308#   2311    2857    2898
ADR2    000434            384#     385     2100#   2215#   2218    2244    2309#   2312
AENV  = 000000            333      344
AENVM = 000000            333      345
AFATAL= 000000            333      336
AMADR1= 000000            333
AMADR2= 000000            333
AMADR3= 000000            333
AMADR4= 000000            333
AMAMS1= 000000            333
AMAMS2= 000000            333
AMAMS3= 000000            333
AMAMS4= 000000            333
AMSGAD= 000000            333      341
AMSGLG= 000000            333      342
AMSGTY= 000000            333      335
AMTYP1= 000000            333
AMTYP2= 000000            333
AMTYP3= 000000            333
AMTYP4= 000000            333
APASS = 000000            333      338
APRIOR= 000000            333
ASLB0   004006           1230     1239#
ASLB1   012246           2634     2646#
ASL0    005602           1637     1646#
ASL1    014422           3113     3125#
ASRB0   004076           1265#
```

```
ASRB1   012342            2672#
ASR0    005672            1672#
ASR1    014516            3151#
ASWREG= 000000             333     346
ATESTN= 000000             333     337
AUNIT = 000000             333     340
AUSWR = 000000             333     347
AVECT1= 000000             333
AVECT2= 000000             333
BICB0   003140            1042    1051#
BICB1   011202            2396    2408#
BIC0    004710            1440    1449#
BIC1    013336            2878    2891#
BRANCH  001170             608     616#
BTWRD   016462            3548    3558#
CBIT    000740             510     517#
CC0     000612             442     445     446     447     448     450     451     452     453     454     455     456     458#
CC1     000656             471     474     475     476     477     478     479     480     482#
CC2     000726             493     497     498     499     500     501     502     503     504     506#
CC3     000774             519     524     525     526     527     528     529     531#
CC4     001044             542     548     549     550     551     552     553     554     556#
CC5     001156             592     596     598     600     602     604#
CC6     001112             569     572     573     574     575     577     578     579     580     582#
CLNZ  = 000254             323#   1795    3282
CMPB0   003000            1011#
CMPB1   011100            2363    2375#
CMP0    004616            1422#
CMP1    013072            2829#
COMB0   003416            1123#
COMB1   011624            2503    2521#
COM0    005212            1529#
COM1    013772            3000#
DOAGN   017024            3643    3648#   3663
DUMMY   000436             386#    387     879     880
EADD0   006744            1883    1915#
EADD1   015770            3368    3422#
EASRB0  004174            1267    1288#
EASRB1  012450            2674    2697#
EASR0   005770            1674    1695#
EASR1   014624            3153    3176#
EBIC0   005044            1451    1481#
EBIC1   013610            2893    2948#
EBTWRD  016740            3560    3619#
ECMPB0  003130            1013    1043#
ECMPB1  011172            2377    2397#
ECMP0   004700            1424    1441#
ECMP1   013326            2831    2879#
EINCB1  011614            2466    2504#
EMODE0  007376            2004    2040#
EMODE2  007534            2054    2082    2085#
EMODE3  007726            2096    2124    2127#
EMODE4  010204            2140    2196    2199#
EMODE5  010434            2210    2254    2257#
EMODE6  010604            2270    2293#
EMODE7  010746            2304    2324#
ENDCC0  000622             457     462#
```

```
ENDCC1  000666            481     486#
ENDCC2  000736            505     510#
ENDCC5  001166            603     608#
ENDJP1  001356            646     670#
ENDJP3  001562            682     726#
ENDJP5  001772            736     780#
ENDJP7  002172            791     829#
ENDJSR  002606            839     945#
ENDPAS  000454            397#   3641
ENEGB0  003612           1160    1175#
ENEGB1  012042           2560    2576#
ENEGO   005406           1566    1581#
ENEGI   014212           3036    3054#
EPSW    007212           1966    1989#
EPSWNO  016452           3522    3549#
EREGS   002714            956     976#
EROLB0  003704           1185    1202#
EROLB1  012140           2589    2607#
EROLO   005500           1591    1608#
EROLI   014314           3067    3086#
ERORB0  003776           1214    1231#
ERORB1  012236           2617    2635#
ERORO   005572           1621    1638#
ERORI   014412           3096    3114#
ERRNM = 000402            324#    459     462#    483     486#    507     510#    532     535#    557     560#    583     586#
                          605     608#    619     622#    624     627#    628     631#    632     635#    636     639#    651
                          654#    660     663#    665     668#    671     674#    687     690#    696     699#    701     704#
                          706     709#    715     718#    720     723#    727     730#    741     744#    746     749#    751
                          754#    756     759#    764     767#    768     771#    773     776#    781     784#    795     798#
                          800     803#    805     808#    810     813#    817     820#    823     826#    830     833#    845
                          848#    850     853#    859     862#    864     867#    869     872#    875     878#    885     888#
                          894     897#    899     902#    904     907#    910     913#    916#    919     922#    925     928#
                          933     936#    940     943#    946     949#    977     980#    991     994#   1020    1023#   1026
                         1029#   1031    1034#   1037    1040#   1044    1047#   1054    1057#   1065    1068#   1072    1075#
                         1089    1092#   1106    1109#   1126    1129#   1136    1139#   1144    1147#   1167    1170#   1176
                         1179#   1194    1197#   1203    1206#   1222    1225#   1232    1235#   1242    1245#   1253    1256#
                         1275    1278#   1289    1292#   1299    1302#   1314    1317#   1323    1326#   1340    1343#   1350
                         1353#   1360    1363#   1371    1374#   1393    1396#   1409    1412#   1431    1434#   1436    1439#
                         1442    1445#   1461    1464#   1468    1471#   1482    1485#   1495    1498#   1512    1515#   1532
                         1535#   1542    1545#   1550    1553#   1573    1576#   1582    1585#   1600    1603#   1609    1612#
                         1629    1632#   1639    1642#   1649    1652#   1660    1663#   1682    1685#   1696    1699#   1706
                         1709#   1721    1724#   1730    1733#   1747    1750#   1757    1760#   1767    1770#   1778    1781#
                         1800    1803#   1809    1812#   1830    1833#   1841    1844#   1871    1874#   1890    1893#   1898
                         1901#   1908    1911#   1916    1919#   1926    1929#   1936    1939#   1943    1946#   1952    1955#
                         1976    1979#   1990    1993#   2011    2014#   2019    2022#   2032    2035#   2041    2044#   2071
                         2074#   2086    2089#   2113    2116#   2128    2131#   2149    2152#   2162    2165#   2172    2175#
                         2182    2185#   2200    2203#   2228    2231#   2238    2241#   2258    2261#   2285    2288#   2294
                         2297#   2317    2320#   2325    2328#   2359    2362#   2365    2368#   2387    2390#   2392    2395#
                         2398    2401#   2411    2414#   2425    2428#   2432    2435#   2449    2452#   2454    2457#   2489
                         2492#   2505    2508#   2524    2527#   2537    2540#   2546    2549#   2568    2571#   2577    2580#
                         2599    2602#   2608    2611#   2626    2629#   2636    2639#   2649    2652#   2661    2664#   2684
                         2687#   2698    2701#   2711    2714#   2727    2730#   2736    2739#   2750    2753#   2761    2764#
                         2771    2774#   2782    2785#   2805    2808#   2840    2843#   2845    2848#   2864    2867#   2870
                         2873#   2880    2883#   2907    2910#   2914    2917#   2923    2926#   2937    2940#   2949    2952#
                         2959    2962#   2980    2983#   3003    3006#   3014    3017#   3022    3025#   3046    3049#   3055
                         3058#   3077    3080#   3087    3090#   3105    3108#   3115    3118#   3128    3131#   3140    3143#
```

```
                        3163    3166#   3177    3180#   3190    3193#   3206    3209#   3215    3218#   3229    3232#   3240
                        3243#   3250    3253#   3261    3264#   3287    3290#   3296    3299#   3315    3318#   3326    3329#
                        3359    3362#   3378    3381#   3385    3388#   3394    3397#   3414    3417#   3423    3426#   3445
                        3448#   3451    3454#   3460    3463#   3472    3475#   3488    3491#   3496    3499#   3501    3504#
                        3509    3512#   3537    3540#   3550    3553#   3569    3572#   3578    3581#   3587    3590#   3594
                        3597#   3600    3603#   3610    3613#   3620    3623#   3634    3637#   3688    3691#   3697    3700#
                        3706    3709#   3716    3719#   3726    3729#   3736    3739#   3747    3750#   3756    3759#   3766
                        3769#   3775    3778#   3784    3787#   3794    3797#
ESOB    016330          3481    3508#
ESUB1   016206          3435    3471#
ESWAB0  006466          1821    1840#
ESWAB1  015344          3305    3325#
ESXT0   006366          1791    1808#
ESXT1   015236          3277    3295#
ETSTB1  011070          2341    2364#
EXOR0   006574          1850    1870#
EXOR1   015464          3338    3358#
GET42   017006          3640    3642#
INCB0   003260          1086#
INCB1   011430          2464#
INC0    005054          1480    1492#
INC1    013620          2947    2956#
JMP1    001262           639     644#
JMP2    001366           668     680#
JMP3    001472           704     710#
JMP4    001572           723     734#
JMP5    001670           754     760#
JMP6    002002           778     789#
JMP7    002104           808     814#
JSRM    002344           849     878#
JSRTST  002202           826     837#
MARK2   000452           396#    881
MODE0   007222          1988    2002#
MODE1   007306          2018    2023#
MODE2   007406          2039    2052#
MODE3   007544          2084    2094#
MODE4   007736          2126    2138#
MODE5   010214          2198    2208#
MODE6   010444          2256    2268#
MODE7   010614          2292    2302#
NBIT    000624           462     469#
NEGB0   003532          1158#
NEGB1   011756          2558#
NEG0    005326          1564#
NEG1    014116          3034#
NEXT    016750          3618    3631#
NOBIT   000546           431     440#
NOP1  = 000260           325#    449
NOTCC   001122           581     590#
POWER   000470           400#    3662
PSW     007120          1964#
PSWN0   016340          3507    3520#
PWRDN   017034           417    3657#   3661
PWRUP   017044          3657    3660#
REGS    002616           918     954#
RETURN  017030          3649#
```

E09


```
ROLB0   003622          1174    1183*
ROLB1   012052          2575    2587*
ROL0    005416          1580    1589*
ROL1    014222          3053    3065*
RORB0   003714          1201    1212*
RORB1   012150          2606    2615*
ROR0    005510          1607    1619*
ROR1    014324          3085    3094*
SBCB0   004336          1337*
SBCB1   012616          2747*
SBC0    006132          1744*
SBC1    014772          3226*
SENVC =  000273           327*  1803    3290
SEVC  =  000263           326*  1412    1476    1859    2443    2927    2943    3347
S0B     016216          3470    3479*
START   000530           422     428*   841     848     862     902     917     2946    3583    3585    3597    3649    3660
SUB0    006754          1914    1923*
SUB1    016000          3421    3433*
SWAB0   006376          1807    1819*
SWAB1   015246          3294    3303*
SXT0    006310          1789*
SXT1    015154          3275*
TEMP    000440           388*    389     710*    711*    712     723*    724     814     821     826*    827     958*    973
                         975    1454    2023    2056    2063*   2068    2079*   2083    2098    2104*   2106*   2110    2117*
                        2121*   2125    2142*   2143    2160    2167    2175*   2176    2195    2212*   2213    2233    2241*
                        2253    2273    2278    2307    2343    2380    2415    2468    2528    2562    2809    2834    2877
                        2895    2969    3156    3194    3233    3371    3376    3392    3437    3439*   3455*   3458    3466*
                        3467*   3469    3524    3603*   3606*   3608    3613*   3615*   3617
TEMP1   000442           390*    391     760*    761     762*    771     776*    777     778*    2024    2057    2099    2144
                        2177    2214    2274    2288*   2291    2306*   2308    2315    2322    2344    2417    2441    2447
                        2529    2591    2619    2653    2676    2810    2833    2843    2855*   2867    2900    2921    2963
                        3007    3038    3279    3341*   3343*   3345*   3346    3356    3401*   3402    3438    3440*   3443
                        3448*   3454*   3525    3545*   3547
TEMP2   000444           392*    393    2025    2026*   2030    2038    2058    2100    2103*   2108    2116*   2123    2145
                        2147    2169    2178    2180    2197    2215    2235    2255    2272*   2275    2282    2309    2379
                        2677    2715    2754    2854*   2855    2874*   2940    3069    3098    3132    3155    3307    3370
                        3372*   3383    3388*   3397*   3417*   3418*   3420    3581
TMB     000450           395*   3678*
TPS     000446           394*   3676
TSTB0   002724           974     988*
TSTB1   010756          2323    2339*
TST0    004514          1390*
TST1    013000          2802*
TYPE    017064           406    3669*
VBIT    000670           486     491*
XOR0    006476          1839    1848*
XOR1    015354          3324    3336*
YESCC   001054           555     567*
ZBIT    001004           530     540*
$APTHD  000430           366     372*   379
$CC0    017126          1311    1347    1357    1718    1754    1764    1887    1972    2724    2758    2768    3203    3237
                        3247    3375    3530    3684*
$CC1    017146          1062    1069    1103    1118    1133    1458    1465    1509    1524    1539    2422    2429    2486
                        2501    2534    2849    2904    2911    2977    2992    3011    3693*
$CC10   017316          1001    1003    1404    1406    1428    1827    1933    2351    2353    2384    2819    2821    2837
                        2853    3312    3442    3752*
```

```
$CC11   017336          1017  1076  1078  1099  1113  1141  1164  1285  1368  1472  1475  1505  1519
                        1547  1570  1692  1775  1805  1862  1867  1895  1986  2437  2439  2474  2499
                        2542  2565  2694  2779  2920  2931  2970  3019  3043  3173  3258  3292  3350
                        3355  3391  3546  3566  3607  3616  3761#
$CC12   017360          1191  1228  1250  1320  1597  1635  1657  1727  2596  2632  2658  2733  3074
                        3111  3137  3212  3771#
$CC13   017400          1096  1172  1502  1578  1949  2472  2573  2851  2967  3051  3457  3780#
$CC17   017420          1983  3493  3543  3789#
$CC2    017166          1377  1784  2788  3267  3702#
$CC3    017206          1116  1522  2499  2990  3419  3711#
$CC4    017230           997   999  1259  1281  1306  1365  1399  1401  1666  1688  1713  1772  1837
                        1957  1979  2347  2349  2459  2482  2667  2690  2719  2776  2813  2815  3146
                        3169  3198  3255  3322  3465  3468  3533  3721#
$CC5    017252          1101  1111  1150  1329  1414  1478  1507  1517  1556  1736  1797  1856  2446
                        2494  2553  2742  2929  2945  2975  2985  3029  3221  3284  3344  3406  3410
                        3575  3731#
$CC7    017274          1198  1226  1257  1279  1604  1633  1664  1686  1903  2603  2630  2665  2688
                        3081  3109  3144  3167  3400  3742#
$CPUOP  000426           348#
$DEVCT  000410           339#
$ENDAD  017014           296  3644#
$ENV    000420           344#
$ENVM   000421           345#  3669
$ETABL  000420           343#   418
$ETEND  000430           355#   378
$FATAL  000402           336#
$HD   = 000003           276    277
$HIBTS  000430           373#
$MAIL   000400           334#   374   378   420
$MBAOR  000432           374#
$MSGAD  000414           341#
$MSGLG  000416           342#
$MSGTY  000400           335#
$PASS   000406           338#  3637#  3638
$PASTM  000436           376#
$SVPC = 001000           294#   299
$SWR  = 160000           276    277#
$SWREG  000422           346#
$TESTN  000404           337#   429   897   971  3648#
$TN   = 000124           276#   328#  436   441   442#  465   470   471#  487   492   493#  513   518
                         519#   536   541   542#  563   568   569#  586   591   592#  612   617   618#
                         640    645   646#  676   681   682#  730   735   736#  785   790   791#  833
                         838    839#  950   955   956#  984   989   990#  1007  1012  1013#  1047  1052
                        1053#  1082  1087  1088#  1119  1124  1125#  1154  1159  1160#  1179  1184  1185#
                        1208  1213  1214#  1235  1240  1241#  1261  1266  1267#  1292  1297  1298#  1333
                        1338  1339#  1386  1391  1392#  1418  1423  1424#  1445  1450  1451#  1488  1493
                        1494#  1525  1530  1531#  1560  1565  1566#  1585  1590  1591#  1615  1620  1621#
                        1642  1647  1648#  1668  1673  1674#  1699  1704  1705#  1740  1745  1746#  1785
                        1790  1791#  1815  1820  1821#  1844  1849  1850#  1877  1882  1883#  1919  1924
                        1925#  1960  1965  1966#  1998  2003  2004#  2048  2053  2054#  2090  2095  2096#
                        2134  2139  2140#  2204  2209  2210#  2264  2269  2270#  2298  2303  2304#  2335
                        2340  2341#  2371  2376  2377#  2404  2409  2410#  2460  2465  2466#  2517  2522
                        2523#  2554  2559  2560#  2583  2588  2589#  2611  2616  2617#  2642  2647  2648#
                        2668  2673  2674#  2704  2709  2710#  2743  2748  2749#  2798  2803  2804#  2825
                        2830  2831#  2887  2892  2893#  2952  2957  2958#  2996  3001  3002#  3030  3035
                        3036#  3061  3066  3067#  3090  3095  3096#  3121  3126  3127#  3147  3152  3153#
```

```
                        3183    3188    3189#   3222    3227    3228#   3271    3276    3277#   3299    3304    3305#   3332
                        3337    3338#   3362    3367    3368#   3429    3434    3435#   3475    3480    3481#   3515    3521
                        3522#   3554    3559    3560#   3632    3633#
$TSTM   000434           375#
$UNIT   000412           340#
$UNITM  000440           377#
$USWR   000424           347#
.       = 017442         284#    290     294     295#    297#    299#    313#    362     363#    365#    367#    379#    381#
                         383#    385#    387#    389#    391#    393#    403#    405#    416#    426#    804     1971#   1972#
                        1974#   1975#   1982#   1983#   1985#   1986#   3529#   3530#   3532#   3533#   3542#   3543#   3545#
                        3546#
.TYPE   = 000004         329#    3641    3662
.$X     = 000430         362#    367
..A     = 016432        1971#   1974#   1982#   1985#   3529#   3532#   3542#   3545#
..B     = 016436        1971#   1972    1974#   1975    1982#   1983    1985#   1986    3529#   3530    3532#   3533    3542#
                        3543    3545#   3546
..C     = 000067        1971#   1974#   1982#   1985#   3529#   3532#   3542#   3545#
```

```
ERROR    307#  459  483  507  532  557  583  605  619  624  628  632  636  651  660
               665  671  687  696  701  706  715  720  727  741  746  751  756  764  768
               773  781  795  800  805  810  817  823  830  845  850  859  864  869  875
               885  894  899  904  910  913  919  925  933  940  946  977  991 1020 1026
              1031 1037 1044 1054 1065 1072 1089 1106 1126 1136 1144 1167 1176 1194 1203
              1222 1232 1242 1253 1275 1289 1299 1314 1323 1340 1350 1360 1371 1393 1409
              1431 1436 1442 1461 1468 1482 1495 1512 1532 1542 1550 1573 1582 1600 1609
              1629 1639 1649 1660 1682 1696 1706 1721 1730 1747 1757 1767 1778 1800 1809
              1830 1841 1871 1890 1898 1908 1916 1926 1936 1943 1952 1976 1990 2011 2019
              2032 2041 2071 2086 2113 2128 2149 2162 2172 2182 2200 2228 2238 2258 2285
              2294 2317 2325 2359 2365 2387 2392 2398 2411 2425 2432 2449 2454 2489 2505
              2524 2537 2546 2568 2577 2599 2608 2626 2636 2649 2661 2684 2698 2711 2727
              2736 2750 2761 2771 2782 2805 2840 2845 2864 2870 2880 2907 2914 2923 2937
              2949 2959 2980 3003 3014 3022 3046 3055 3077 3087 3105 3115 3128 3140 3163
              3177 3190 3206 3215 3229 3240 3250 3261 3287 3296 3315 3326 3359 3378 3385
              3394 3414 3423 3445 3451 3460 3472 3488 3496 3501 3509 3537 3550 3569 3578
              3587 3594 3600 3610 3620 3634 3688 3697 3706 3716 3726 3736 3747 3756 3766
              3775 3784 3794
HLT      304#  458  482  506  531  556  582  604  619  624  628  632  636  651  659
               665  670  687  695  701  706  715  720  726  741  745  751  756  764  767
               773  780  795  800  805  810  817  823  829  844  850  858  864  869  875
               884  904  909  913  919  925  933  940  945  976  991 1020 1026 1031 1037
              1043 1054 1065 1072 1089 1106 1126 1136 1144 1167 1175 1194 1202 1222 1231
              1242 1253 1275 1298 1299 1314 1323 1340 1350 1360 1371 1393 1409 1431 1436
              1441 1461 1468 1481 1495 1512 1532 1542 1550 1573 1581 1600 1608 1629 1638
              1649 1660 1682 1695 1706 1721 1730 1747 1757 1767 1778 1800 1808 1830 1840
              1870 1890 1898 1908 1915 1926 1936 1943 1952 1976 1989 2011 2019 2032 2040
              2070 2085 2112 2127 2149 2162 2171 2182 2199 2228 2237 2257 2284 2293 2317
              2324 2359 2364 2387 2392 2397 2411 2425 2432 2449 2454 2489 2504 2524 2537
              2546 2568 2576 2599 2607 2626 2635 2649 2661 2684 2697 2711 2727 2736 2750
              2761 2771 2782 2805 2840 2845 2864 2870 2879 2907 2914 2923 2937 2948 2959
              2980 3003 3014 3022 3046 3054 3077 3086 3105 3114 3128 3140 3163 3176 3190
              3206 3215 3229 3240 3250 3261 3287 3295 3315 3325 3358 3378 3385 3394 3414
              3422 3445 3451 3460 3471 3488 3496 3501 3508 3537 3549 3569 3578 3587 3594
              3600 3610 3619 3634 3687 3696 3705 3715 3725 3735 3746 3755 3765 3774 3783
              3793
HLT1     306#  893  899
MFPS     304# 1973 1984 3531 3544
MTPS     302# 1970 1981 3528 3541
NWTEST   308#  436  465  487  513  536  563  586  612  640  676  730  785  833  950
               984 1007 1047 1082 1119 1154 1179 1208 1235 1261 1292 1333 1386 1418 1445
              1488 1525 1560 1585 1615 1642 1668 1699 1740 1785 1815 1844 1877 1919 1960
              1998 2048 2090 2134 2204 2264 2298 2335 2371 2404 2460 2517 2554 2583 2611
              2642 2668 2704 2743 2798 2825 2887 2952 2996 3030 3061 3090 3121 3147 3183
              3222 3271 3299 3332 3362 3429 3475 3515 3554
SEQCHK   311#  440  469  491  517  540  567  590  616  644  680  734  789  837  954
               988 1011 1051 1086 1123 1158 1183 1212 1239 1265 1296 1337 1390 1422 1449
              1492 1529 1564 1589 1619 1646 1672 1703 1744 1789 1819 1848 1881 1923 1964
              2002 2052 2094 2138 2208 2268 2302 2339 2375 2408 2464 2521 2558 2587 2615
              2646 2672 2708 2747 2802 2829 2891 2956 3000 3034 3065 3094 3125 3151 3187
              3226 3275 3303 3336 3366 3433 3479 3520 3558 3631
SEQUEN   310#  441  470  492  518  541  568  591  617  645  681  735  790  838  955
               989 1012 1052 1087 1124 1159 1184 1213 1240 1266 1297 1338 1391 1423 1450
              1493 1330 1565 1590 1620 1647 1673 1704 1745 1790 1820 1849 1882 1924 1965
              2003 2053 2095 2139 2209 2269 2303 2340 2376 2409 2465 2522 2559 2588 2616
              2647 2673 2709 2748 2803 2830 2892 2957 3001 3035 3066 3095 3126 3152 3188
```

```
DVKAAB  MACY11 30(1046)  25-OCT-77  13:11  PAGE 114
DVKAAB.P11     25-OCT-77 13:09          CROSS REFERENCE TABLE -- MACRO NAMES                                    SEQ 0112

          3227    3276    3304    3337    3367    3434    3480    3521    3559    3632
STARS      266#     286     292     310#     332     359     361     368     436     438     465     467     487     489     513
            515     536     538     563     565     586     588     612     614     640     642     676     678     730     732
            785     787     833     835     950     952     984     986    1007    1009    1047    1049    1082    1084    1119
           1121    1154    1156    1179    1181    1208    1210    1235    1237    1261    1263    1292    1294    1333    1335
           1386    1388    1418    1420    1445    1447    1488    1490    1525    1527    1560    1562    1585    1587    1615
           1617    1642    1644    1668    1670    1699    1701    1740    1742    1785    1787    1815    1817    1844    1846
           1877    1879    1919    1921    1960    1962    1998    2000    2048    2050    2090    2092    2134    2136    2204
           2206    2264    2266    2298    2300    2335    2337    2371    2373    2404    2406    2460    2462    2517    2519
           2554    2556    2583    2585    2611    2613    2642    2644    2668    2670    2704    2706    2743    2745    2798
           2800    2825    2827    2887    2889    2952    2954    2996    2998    3030    3032    3061    3063    3090    3092
           3121    3123    3147    3149    3183    3185    3222    3224    3271    3273    3299    3301    3332    3334    3362
           3364    3429    3431    3475    3477    3515    3517    3554    3556    3652
$C17        301#     654     690     853     888
$NWTST      309#     436     465     487     513     536     563     586     612     640     676     730     785     833     950
            984    1007    1047    1082    1119    1154    1179    1208    1235    1261    1292    1333    1386    1418    1445
           1488    1525    1560    1585    1615    1642    1668    1699    1740    1785    1815    1844    1877    1919    1960
           1998    2048    2090    2134    2204    2264    2298    2335    2371    2404    2460    2517    2554    2583    2611
           2642    2668    2704    2743    2798    2825    2887    2952    2996    3030    3061    3090    3121    3147    3183
           3222    3271    3299    3332    3362    3429    3475    3515    3554
.HEADE     266#
.$ACTI     266#     290
.$APTB     266#     330
.$APTH     266#     357


. ABS.   017442     000

 ERRORS DETECTED:  0

 DVKAAB.BIN,DVKAAB.LST/CRF/SOL/NL:TOC=DVKAAB.P11
 RUN-TIME:  9 10 1 SECONDS
 RUN-TIME RATIO: 359/21=16.8
 CORE USED:  9K  (17 PAGES)
```