LSI-11

BASIC INSTRUCTION TESTS

CVKAAC0

AH-8187C-MC

COPYRIGHT 75-78

FICHE 1 OF 1

JAN 1979

digital

MADE IN USA

```
.REM    %
```

IDENTIFICATION

PRODUCT CODE:	AC-8186C-MC
PRODUCT NAME:	CVKAACO LSI-11 BSC INST
PRODUCT DATE:	01-NOVEMBER-1978
MAINTAINER:	DIAGNOSTIC ENGINEERING
AUTHOR:	PERVEZ A. ZAKI
MODIFIED BY:	BARRY SUSSMAN 01-NOV-77
	BARRY SUSSMAN 01-NOV-78

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1975, 1978 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

CONTENTS

1. ABSTRACT
2. REQUIREMENTS
2.1 EQUIPMENT
2.2 STORAGE
2.3 PRELIMINARY PROGRAMS
3. LOADING PROCEDURE
4. STARTING PROCEDURE
4.1 STARTING ADDRESS
4.2 PROGRAM AND/OR OPERATOR ACTION
5. OPERATING PROCEDURE
6. ERRORS
6.1 ERROR REPORTING
6.2 ERROR RECOVERY
7. RESTRICTIONS
8. MISCELLANEOUS
8.1 EXECUTION TIME
8.2 STACK POINTER
8.3 PASS COUNTER
8.4 TEST NUMBER
8.5 POWER FAIL
9. PROGRAM DESCRIPTION

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 102] AND RESTART FROM 450 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 450, IN WHICH CASE THE PROGRAM WILL HALT
AT LOCATION 464 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) PLACE LTC SWITCH IN OFF POSITION.
2) LOAD PROGRAM INTO MEMORY USING ABS LOADER.
3) TYPE 200G [THERE ARE NO SWITCH SETTINGS AND THIS DIAGNOSTIC DOES NOT USE SOFTWARE SWITCH LOCATION $SWREG]
4) 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 117] 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 PLACING A 20000 IN LOCATION 116.

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 100]. 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 102] 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 450 (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)

WHEN RUNNING UNDER APT IN A SCRIPT, THE FIRST PASS RUN TIME
AND SUBSEQUENT PASS RUN TIMES ARE ONE (1) SECOND.

8.2 STACK POINTER

STACK IS INITIALLY SET TO 450

8.3 PASS COUNT

A 16 BIT LOCATION "$PASS" (I.E. LOCATION 104) 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 102) 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 450 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.
                                %
```

```
                                        .ABS
                                        ;; LSI-11 MACRO INSTRUCTION EXERCISER
                                        .NLIST  MC,MD,CND
                                        .LIST   ME
                                .TITLE  CVKAAC
                                ;*COPYRIGHT (C) 1975, 1978
                                ;*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.
                                ;*
                                $TN=1
        000001                  $SWR=160000     ;;HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYPOUT
        160000
```

```
        000000                          .=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  016734                          $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
        000076                          .=76
        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   =101
```

```
                                .SBTTL  APT MAILBOX-ETABLE

                                ;;*******************************************************************
                                .EVEN
000076                          $MAIL:                          ;;APT MAILBOX
000076  000000                  $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
000100  000000                  $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
000102  000000                  $TESTN: .WORD   ATESTN          ;;TEST NUMBER
000104  000000                  $PASS:  .WORD   APASS           ;;PASS COUNT
000106  000000                  $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
000110  000000                  $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
000112  000000                  $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
000114  000000                  $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
000116                          $ETABLE:                        ;;APT ENVIRONMENT TABLE
000116     000                  $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
000117     000                  $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
000120  000000                  $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
000122  000000                  $USWR:  .WORD   AUSWR           ;;USER SWITCHES
000124  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
000126                          $ETEND:
                                .MEXIT
                                .SBTTL  APT PARAMETER BLOCK

                                ;;*******************************************************************
                                ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
                                ;;*******************************************************************
        000126                          .$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  000126                          $APTHDR ;;POINT TO APT HEADER BLOCK
        000126                          .=.$X   ;;RESET LOCATION COUNTER
                                ;;*******************************************************************
                                ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                                ;INTERFACE SPEC.

000126                          $APTHD:
000126  000000                  $HIBTS: .WORD   0               ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
000130  000076                  $MBADR: .WORD   $MAIL           ;;ADDRESS OF APT MAILBOX (BITS 0-15)
000132  000001                  $TSTM:  .WORD   1               ;;RUN TIM OF LONGEST TEST
000134  000001                  $PASTM: .WORD   1               ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
000136  000000                  $UNITM: .WORD                   ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
000140  000014                          .WORD   $ETEND-$MAIL/2  ;;LENGTH MAILBOX-ETABLE(WORDS)
```

```
            000126                          .=$APTHD
000126                                      ADR:
            000130                          .=ADR+2
000130                                      ADR1:
            000132                          .=ADR1+2
000132                                      ADR2:
            000134                          .=ADR2+2
000134                                      DUMMY:
            000136                          .=DUMMY+2
000136                                      TEMP:
            000140                          .=TEMP+2
000140                                      TEMP1:
            000142                          .=TEMP1+2
000142                                      TEMP2:
            000144                          .=TEMP2+2
000144      177564                          TPS:     177564          ; OUTPUT TTY STATUS REGISTER
000146      177566                          TPB:     177566          ; OUTPUT BUFFER
000150      006402                  MARK2:  MARK     2
000152      005015  042440  042116  ENDPAS: .ASCIZ   <15><12>'' END PASS''
000160      050040  051501  000123
                                            .EVEN
000166      005015  047520  042527  POWER:  .ASCIZ   <15><12>/POWER/
000174      000122
                                            .EVEN
000176      000023                          .BLKW    19.

            000020                          .=20
000020      017004                          TYPE
000022      000000                          0
```

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

        000200                          .=200
000200  012737  016754  000024          MOV     #PWRDN,@#24     ; SERVICE POWER DOWN ROUTINE ON ANY FUTURE POWER DOWN
000206  012700  000116                  MOV     #$ETABLE,R0
000212  005040                  2$:     CLR     -(R0)           ;START CLEANING THE STACK
000214  020027  000076                  CMP     R0,#$MAIL       ;FOR INITIALIZATION
000220  101374                          BHI     2$
000222  000167  000222                  JMP     START
                                ;       -------------------------

        000450                          .=450

000450  012706  000450          START:  MOV     #START,SP       ; SET THE STACK POINTER
000454  012705  000102                  MOV     #$TESTN,R5      ; PLACE THE ADDRESS OF LOCATION $TESTN IN R5
000460  005715                          TST     (R5)            ; CHECK THE SEQUENCE COUNTER
000462  001401                          BEQ     NOBIT           ; IF THIS IS THE STARTING OF THE TEST THEN
                                                                ; GO TO NOBIT TEST
000464  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
                                ;;*******************************************************************************

000466                          NOBIT:
000466  021527  000000                  CMP     (R5),#0
000472  001017                          BNE     CC0                     ; IF IN WRONG SEQUENCE GO TO HALT AT END OF THE TEST
000474  005215                  1$:     INC     (R5)
000476  000257                          CCC                             ; ZERO CONDITION CODES, NZVC=0000
000500  103414                          BCS     CC0
000502  102413                          BVS     CC0
000504  001412                          BEQ     CC0
000506  100411                          BMI     CC0
000510  000260                          NOP1                            ; CHECK NOP1 INSTRUCTION I.E. OP-CODE 260
000512  103407                          BCS     CC0
000514  102406                          BVS     CC0
000516  001405                          BEQ     CC0
000520  100404                          BMI     CC0
000522  002403                          BLT     CC0
000524  003402                          BLE     CC0
000526  101401                          BLOS    CC0
000530  101004                          BHI     ENDCC0
000532                          CC0:
000532  012745  000001                  MOV     #1,-(R5)
000536  005245                          INC     -(R5)
000540  000000                          HALT                            ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
000542  102000                  ENDCC0: BVC     NBIT


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


000544                          NBIT:
000544  021527  000001                  CMP     (R5),#1
000550  001012                          BNE     CC1                     ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THETEST
000552  005215                  1$:     INC     (R5)
000554  000270                          SEN                             ; NBIT IS SET, NZVC=1000
000556  100007                          BPL     CC1
000560  001406                          BEQ     CC1
000562  002005                          BGE     CC1
000564  003004                          BGT     CC1
000566  103403                          BCS     CC1
000570  101402                          BLOS    CC1
000572  103401                          BLO     CC1
000574  003404                          BLE     ENDCC1
000576                          CC1:
000576  012745  000002                  MOV     #2,-(R5)
000602  005245                          INC     -(R5)
000604  000000                          HALT                            ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
000606  001000                  ENDCC1: BNE     VBIT
```

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

000610                          VBIT:
000610  021527  000002                  CMP     (R5),#2
000614  001014                          BNE     CC2                     ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
000616  005215                  1$:     INC     (R5)
000620  000270                          SEN
000622  000262                          SEV                             ; V AND N BIT SET, NZVC = 1010
000624  102010                          BVC     CC2
000626  001407                          BEQ     CC2
000630  100006                          BPL     CC2
000632  103405                          BCS     CC2
000634  002404                          BLT     CC2
000636  003403                          BLE     CC2
000640  101402                          BLOS    CC2
000642  103401                          BLO     CC2
000644  003004                          BGT     ENDCC2
000646                          CC2:
000646  012745  000003                  MOV     #3,-(R5)
000652  005245                          INC     -(R5)
000654  000000                          HALT                            ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
000656  002000                  ENDCC2: BGE     CBIT


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

000660                          CBIT:
000660  021527  000003                  CMP     (R5),#3
000664  001013                          BNE     CC3                     ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
000666  005215                  1$:     INC     (R5)
000670  000270                          SEN
000672  000262                          SEV
000674  000261                          SEC                             ; C, V, AND N BITS ARE SET, NZVC=1011
000676  001406                          BEQ     CC3
000700  100005                          BPL     CC3
000702  102004                          BVC     CC3
000704  002403                          BLT     CC3
000706  003402                          BLE     CC3
000710  101001                          BHI     CC3
000712  002004                          BGE     ZBIT
000714                          CC3:
000714  012745  000004                  MOV     #4,-(R5)
000720  005245                          INC     -(R5)
000722  000000                          HALT                            ; ONE OF THE ABOVE BRANCHES FAILED
                                                                        ; OR WRONG SEQUENCE
```

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

000724                          ZBIT:
000724  021527  000004                  CMP     (R5),#4
000730  001015                          BNE     CC4             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
000732  005215                          INC     (R5)
000734  000270                          SEN
000736  000262                          SEV
000740  000261                          SEC
000742  000264                          SEZ                     ; ALL BITS SET, NZVC=1111
000744  001007                          BNE     CC4
000746  100006                          BPL     CC4
000750  102005                          BVC     CC4
000752  103004                          BCC     CC4
000754  002403                          BLT     CC4
000756  003002                          BGT     CC4
000760  101001                          BHI     CC4
000762  001404                          BEQ     YESCC
000764                          CC4:
000764  012745  000005                  MOV     #5,-(R5)
000770  005245                          INC     -(R5)
000772  000000                          HALT                    ; ONE OF THE ABOVE BRANCHES FAILED
                                                                ; OR WRONG SEQUENCE


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

000774                          YESCC:
000774  021527  000005                  CMP     (R5),#5
001000  001014                          BNE     CC6             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
001002  005215                          INC     (R5)
001004  000277                          SCC                     ; NZVC=1111
001006  100011                          BPL     CC6
001010  001010                          BNE     CC6
001012  102007                          BVC     CC6
001014  103006                          BCC     CC6
001016  000240                          NOP                     ; CHECK NOP INSTRUCTION
001020  100004                          BPL     CC6
001022  001003                          BNE     CC6
001024  102002                          BVC     CC6
001026  103001                          BCC     CC6
001030  101404                          BLOS    NOTCC
001032                          CC6:
001032  012745  000006                  MOV     #6,-(R5)
001036  005245                          INC     -(R5)
001040  000000                          HALT                    ; SCC OR A BRANCH FAILED, OR WRONG SEQUENCE
```

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

001042                          NOTCC:
001042  021527  000006                  CMP     (R5),#6
001046  001013                          BNE     CC5             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
001050  005215                          INC     (R5)
001052  000277                          SCC                     ; NZVC=1111
001054  000241                          CLC                     ; NZVC=1110
001056  103407                          BCS     CC5
001060  000242                          CLV                     ; NZVC=1100
001062  102405                          BVS     CC5
001064  000244                          CLZ                     ; NZVC=1000
001066  001403                          BEQ     CC5
001070  000250                          CLN                     ; NZVC=0000
001072  100401                          BMI     CC5
001074  101004                          BHI     ENDCC5
001076                          CC5:
001076  012745  000007                  MOV     #7,-(R5)
001102  005245                          INC     -(R5)
001104  000000                          HALT                    ; ONE OF THE ABOVE CLEARS FAILED OR WRONG SEQUENCE
001106  100000                  ENDCC5: BPL     BRANCH


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

001110                          BRANCH:
001110  021527  000007                  CMP     (R5),#7
001114  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
001116  012745  000010                  MOV     #10,-(R5)
001122  005245                          INC     -(R5)
001124  000000                          HALT
001126  005215                  1$:     INC     (R5)
001130  000416                          BR      4$              ; CHECK BRANCH FORWARD AND BACKWARD
001132  012745  000011                  MOV     #11,-(R5)
001136  005245                          INC     -(R5)
001140  000000                          HALT                    ; FORWARD BRANCH FAILED
001142  000404                  2$:     BR      3$
001144  012745  000012                  MOV     #12,-(R5)
001150  005245                          INC     -(R5)
001152  000000                          HALT                    ; FORWARD BRANCH FAILED
001154  000411                  3$:     BR      5$
001156  012745  000013                  MOV     #13,-(R5)
001162  005245                          INC     -(R5)
001164  000000                          HALT                    ; FORWARD BRANCH FAILED
001166  000765                  4$:     BR      2$
001170  012745  000014                  MOV     #14,-(R5)
001174  005245                          INC     -(R5)
001176  000000                          HALT                    ; BACKWARD BRANCH FAILED
001200  000400                  5$:     BR      JMP1
```

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

                                JMP1:
001202                                  CMP     (R5),#10
001202  021527  000010                  BNE     ENDJP1          ; IF IN WRONG SEQUENCE GO TO HALT AT THE END OF THE TEST
001206  001033                  1$:     INC     (R5)
001210  005215                          MOV     #2$,R0          ; TEST JUMP INSTRUCTION MODE 1
001212  012700  001232                  SCC
001216  000277                          JMP     (R0)
001220  000110                          MOV     #15,-(R5)
001222  012745  000015                  INC     -(R5)
001226  005245                          HALT                    ; JUMP INSTRUCTION FAILED
001230  000000                  2$:
001232                                  BPL     3$
001232  100003                          BNE     3$
001234  001002                          BVC     3$
001236  102001                          BCS     4$
001240  103404                  3$:
001242                                  MOV     #16,-(R5)
001242  012745  000016                  INC     -(R5)
001246  005245                          HALT                    ; WRONG CC
001250  000000                  4$:     CMP     R0,#2$
001252  020027  001232                  BEQ     5$              ; CONTINUE IF R0 IS OK
001256  001404                          MOV     #17,-(R5)
001260  012745  000017                  INC     -(R5)
001264  005245                          HALT
001266  000000                  5$:     MOV     #JMP2,R0        ; TEST JUMP INSTRUCTION MODE 1
001270  012700  001306                  JMP     (R0)
001274  000110                  ENDJP1:
001276                                  MOV     #20,-(R5)
001276  012745  000020                  INC     -(R5)
001302  005245                          HALT                    ; JUMP INSTRUCTION FAILED OR WRONG SEQUENCE
001304  000000


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

                                JMP2:
001306                                  CMP     (R5),#11
001306  021527  000011                  BNE     ENDJP3          ; IF IN WRONG SEQUENCE GO TO HALT AT THE END OF TEST
001312  001073                          INC     (R5)
001314  005215                          MOV     #3$,R0          ; TEST JUMP INSTRUCTION MODE 2
001316  012700  001336                  SCC
001322  000277                          JMP     (R0)+
001324  000120                          MOV     #21,-(R5)
001326  012745  000021                  INC     -(R5)
001332  005245                          HALT                    ; JUMP INSTRUCTION FAILED
001334  000000                  3$:
001336                                  BPL     4$
001336  100003                          BNE     4$
001340  001002                          BVC     4$
001342  102001                          BCS     5$
001344  103404                  4$:
001346
```

```
001346  012745  000022                  MOV     #22,-(R5)
001352  005245                          INC     -(R5)
001354  000000                          HALT                    ; WRONG CC
001356  020027  001340          5$:     CMP     R0,#3$+2        ; IS THERE AUTO INC.?
001362  001404                          BEQ     6$
001364  012745  000023                  MOV     #23,-(R5)
001370  005245                          INC     -(R5)
001372  000000                          HALT                    ; MODE 2 FAILED FOR JMP INSTRUCTION
001374  012700  001412          6$:     MOV     #JMP3,R0        ; TEST JUMP INSTRUCTION MODE 2
001400  000120                          JMP     (R0)+
001402  012745  000024                  MOV     #24,-(R5)
001406  005245                          INC     -(R5)
001410  000000                          HALT                    ; JUMP INSTRUCTION FAILED

001412  012767  001446  176516  JMP3:   MOV     #3$,TEMP        ; TEST JUMP INSTRUCTION MODE 3
001420  012767  001466  176512          MOV     #4$,TEMP+2      ;
001426  012700  000136                  MOV     #TEMP,R0
001432  000277                          SCC
001434  000130                          JMP     @(R0)+          ;
001436  012745  000025                  MOV     #25,-(R5)
001442  005245                          INC     -(R5)
001444  000000                          HALT                    ; JUMP INSTRUCTION FAILED
001446  027067  000000  000012  3$:     CMP     @(R0),4$        ; IS THERE AUTO INC.?
001454  001404                          BEQ     4$
001456  012745  000026                  MOV     #26,-(R5)
001462  005245                          INC     -(R5)
001464  000000                          HALT                    ; JMP INSTRUCTION FAILED IN MODE 2
001466  012767  001512  176442  4$:     MOV     #JMP4,TEMP      ; TEST JUMP INSTRUCTION MODE 3
001474  012700  000136                  MOV     #TEMP,R0
001500  000130                          JMP     @(R0)+
001502                          ENDJP3:
001502  012745  000027                  MOV     #27,-(R5)
001506  005245                          INC     -(R5)
001510  000000                          HALT                    ; JUMP ERROR OR WRONG SEQUENCE
```

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

001512                          JMP4:
001512  021527  000012                  CMP     (R5),#12
001516  001075                          BNE     ENDJP5          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
001520  005215                          INC     (R5)
001522  012700  001544                  MOV     #3$,R0          ; TEST JUMP INSTRUCTION MODE 4
001526  000277                          SCC
001530  000140                          JMP     -(R0)
001532  012745  000030                  MOV     #30,-(R5)
001536  005245                          INC     -(R5)
001540  000000                          HALT                    ; JUMP INSTRUCTION FAILED
001542  000404                          BR      4$              ; JUMP SHOULD LAND HERE
001544                          3$:
001544  012745  000031                  MOV     #31,-(R5)
001550  005245                          INC     -(R5)
001552  000000                          HALT                    ; NO AUTO DECREMENT FROM JMP4
001554  022700  001542          4$:     CMP     #3$-2,R0        ; CHECK R0
001560  001404                          BEQ     5$
001562  012745  000032                  MOV     #32,-(R5)
001566  005245                          INC     -(R5)
001570  000000                          HALT
001572  012700  001612          5$:     MOV     #JMP5+2,R0      ; TEST JUMP INSTRUCTION MODE 4
001576  000140                          JMP     -(R0)
001600  012745  000033                  MOV     #33,-(R5)
001604  005245                          INC     -(R5)
001606  000000                          HALT                    ; JUMP INSTRUCTION FAILED

001610  012767  001642  176322  JMP5:   MOV     #3$,TEMP1       ; TEST JUMP INSTRUCTION MODE 5
001616  012700  000140                  MOV     #TEMP1,R0
001622  012767  001652  176306          MOV     #4$,TEMP1-2     ;
001630  000150                          JMP     @-(R0)
001632  012745  000034                  MOV     #34,-(R5)
001636  005245                          INC     -(R5)
001640  000000                          HALT                    ; JUMP INSTRUCTION FAILED
001642                          3$:
001642  012745  000035                  MOV     #35,-(R5)
001646  005245                          INC     -(R5)
001650  000000                          HALT                    ; ERROR, NO AUTO DECREMENT
001652  022700  000136          4$:     CMP     #TEMP1-2,R0     ; CHECK R0
001656  001404                          BEQ     5$
001660  012745  000036                  MOV     #36,-(R5)
001664  005245                          INC     -(R5)
001666  000000                          HALT                    ; JUMP ONSTRUCTION FAILED IN MODE 5
001670  012767  001642  176242  5$:     MOV     #3$,TEMP1       ; TEST JUMP INSTRUCTION MODE 5
001676  012700  000140                  MOV     #TEMP1,R0
001702  012767  001722  176226          MOV     #JMP6,TEMP1-2   ;
001710  000150                          JMP     @-(R0)
001712                          ENDJP5:
001712  012745  000037                  MOV     #37,-(R5)
001716  005245                          INC     -(R5)
001720  000000                          HALT                    ; JUMP ERROR OR WRONG SEQUENCE

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

```
                                        ;*TEST: 13      TEST JMP INSTRUCTION FOR MODE 6 AND 7
                                        ;;*****************************************************************
                                        JMP6:
001722                                          CMP     (R5),#13
001722  021527  000013                          BNE     ENDJP7          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
001726  001071                                  INC     (R5)
001730  005215                                  MOV     #1$+6,R3
001732  012703  001760                          JMP     -6(R3)
001736  000163  177772                          MOV     #40,-(R5)
001742  012745  000040                          INC     -(R5)
001746  005245                                  HALT                    ; JUMP INSTRUCTION FAILED
001750  000000                          1$:     CMP     R3,#1$+6        ; CHECK R3
001752  020327  001760                          BEQ     2$
001756  001404                                  MOV     #41,-(R5)
001760  012745  000041                          INC     -(R5)
001764  005245                                  HALT                    ; WRONG VALUE IN REGISTER AFTER JUMP MODE 6
001766  000000                                                          ; OR JUMP INSTRUCTION FAILED IN MODE 6
                                        2$:     JMP     3$-.-4(PC)      ; TEST JUMP INSTRUCTION MODE 6
001770  000167  000010                          MOV     #42,-(R5)
001774  012745  000042                          INC     -(R5)
002000  005245                                  HALT                    ; JUMP INSTRUCTION FAILED
002002  000000                          3$:     MOV     #JMP7,R3        ; JUMP SHOULD LAND HERE
002004  012703  002024                          JMP     0(R3)
002010  000163  000000                          MOV     #43,-(R5)
002014  012745  000043                          INC     -(R5)
002020  005245                                  HALT                    ; JUMP INSTRUCTION FAILED
002022  000000
                                        JMP7:   MOV     #TEMP,R3
002024  012703  000136                          MOV     #1$,(R3)
002030  012713  002050                          JMP     @(R3)
002034  000173  000000                          MOV     #44,-(R5)
002040  012745  000044                          INC     -(R5)
002044  005245                                  HALT                    ; JUMP INSTRUCTION FAILED
002046  000000                          1$:     MOV     #3$,(R3)        ; TEST JUMP INSTRUCTION MODE 7
002050  012713  002074                          MOV     #TEMP-4,R0
002054  012700  000132                          JMP     @4(R0)
002060  000170  000004                          MOV     #45,-(R5)
002064  012745  000045                          INC     -(R5)
002070  005245                                  HALT                    ; JUMP INSTRUCTION FAILED
002072  000000                          3$:     MOV     #JSRTST,TEMP    ; CONTINUE
002074  012767  002122  176034                  MOV     #TEMP,R0
002102  012700  000136                          JMP     @0(R0)
002106  000170  000000                  ENDJP7:
002112                                          MOV     #46,-(R5)
002112  012745  000046                          INC     -(R5)
002116  005245                                  HALT                    ; JUMP ERROR OR SEQUENCE ERROR
002120  000000
```

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

002122                          JSRTST:
002122  021527  000014                  CMP     (R5),#14
002126  001177                          BNE     ENDJSR          ; IF IN WRONG SEQUENCE GO TO HALT AT THE END OF THE TEST
002130  005215                          INC     (R5)
002132  012706  000450                  MOV     #START,SP       ; SET UP STACK POINTER.
002136  000277                          SCC
002140  004767  000026                  JSR     PC,3$
002144                          1$:
002144  012745  000047                  MOV     #47,-(R5)
002150  005245                          INC     -(R5)
002152  000000                          HALT                    ; JSR INSTRUCTION FAILED
002154  022706  000450          2$:     CMP     #START,SP       ; HAS SP BEEN RESTORED?
002160  001441                          BEQ     JSRM
002162  012745  000050                  MOV     #50,-(R5)
002166  005245                          INC     -(R5)
002170  000000                          HALT                    ; SP WAS NOT RESTORED BY RTS INSTRUCTION
002172                          3$:
002172  100003                          BPL     4$
002174  001002                          BNE     4$
002176  102001                          BVC     4$
002200  103404                          BCS     5$
002202                          4$:
002202  012745  000051                  MOV     #51,-(R5)
002206  005245                          INC     -(R5)
002210  000000                          HALT                    ; WRONG CC
002212  022706  000446          5$:     CMP     #START-2,SP     ; WAS THE SP EFFECTED?
002216  001404                          BEQ     6$
002220  012745  000052                  MOV     #52,-(R5)
002224  005245                          INC     -(R5)
002226  000000                          HALT                    ; WRONG SP AFTER EXECUTION OF JSR INSTRUCTION
002230  022716  002144          6$:     CMP     #1$,(SP)        ; IS THE RETURN ADDRESS =1$
002234  001404                          BEQ     7$
002236  012745  000053                  MOV     #53,-(R5)
002242  005245                          INC     -(R5)
002244  000000                          HALT                    ; SP DID NOT HAVE CORRECT RETURN ADDRESS
                                                                ; AFTER EXECUTION OF JSR INSTRUCTION
002246  012716  002154          7$:     MOV     #2$,(SP)        ; SET 2$ AS THE RETURN ADDRESS
002252  000207                          RTS     PC
002254  012745  000054                  MOV     #54,-(R5)
002260  005245                          INC     -(R5)
002262  000000                          HALT                    ; RTS INSTRUCTION FAILED
002264  010546                  JSRM:   MOV     R5,-(SP)        ; MOV R5 TO STACK
002266  016746  175642                  MOV     DUMMY,-(SP)
002272  016746  175636                  MOV     DUMMY,-(SP)
002276  016746  175646                  MOV     MARK2,-(SP)     ; STORE MARK 2 ON THE STACK.
002302  010503                          MOV     R5,R3           ; SAVE R5 IN R3
002304  004467  000130                  JSR     R4,10$
002310                          1$:
002310  012745  000055                  MOV     #55,-(R5)
002314  005245                          INC     -(R5)
002316  000000                          HALT                    ; JSR INSTRUCTION FAILED
002320                          2$:
```

```
002320  100003                          BPL     3$
002322  001002                          BNE     3$
002324  102001                          BVC     3$
002326  103404                          BCS     4$
002330                          3$:
002330  012743  000056                  MOV     #56,-(R3)
002334  005243                          INC     -(R3)
002336  000000                          HALT                            ; WRONG CC
002340  022705  000102          4$:     CMP     #$TESTN,R5
002344  001404                          BEQ     5$
002346  012743  000057                  MOV     #57,-(R3)
002352  005243                          INC     -(R3)
002354  000000                          HALT                            ; MARK INSRUCTION FAILED
002356  022706  000450          5$:     CMP     #START,SP
002362  001404                          BEQ     6$
002364  012745  000060                  MOV     #60,-(R5)
002370  005245                          INC     -(R5)
002372  000000                          HALT                            ; MARK INSTRUCTION FAILED
002374  012701  002502          6$:     MOV     #12$,R1                 ; PLACE THE ADDRESS OF 12$ IN R1
002400  004011                          JSR     R0,(R1)                 ; GO TO TAG 12$
002402                          7$:
002402  012745  000061                  MOV     #61,-(R5)
002406  005245                          INC     -(R5)
002410  000000                          HALT                            ; JSR INSTRUCTION FAILED
002412  012745  000062                  MOV     #62,-(R5)
002416  005245                          INC     -(R5)
002420  000000                          HALT                            ; RTS BROUGHT THE PROGRAM BACK IN WRONG
                                                                        ; PLACE
002422  022706  000450          8$:     CMP     #START,SP
002426  001443                          BEQ     REGS
002430  012745  000063                  MOV     #63,-(R5)
002434  005245                          INC     -(R5)
002436  000000                          HALT                            ; STACK POINTER WAS NOT RESET

002440  020427  002310          10$:    CMP     R4,#1$                  ; IS THE RETURN ADDRESS =1$ ?
002444  001404                          BEQ     11$
002446  012745  000064                  MOV     #64,-(R5)
002452  005245                          INC     -(R5)
002454  000000                          HALT                            ; WRONG RETURN ADDRESS IN UNKAGE REGISTER R4
002456  010605                  11$:    MOV     SP,R5
002460  005725                          TST     (R5)+                   ; SET UP ADDRESS IN R5 AT MARK 2 INSTRUCTION
002462  012716  002320                  MOV     #2$,(SP)                ; SET RETURN ADDRESS =2$
002466  000277                          SCC
002470  000205                          RTS     R5                      ; RETURN USING R5 AND IN-TURN USING MARK INSTRUCTION
002472  012745  000065                  MOV     #65,-(R5)
002476  005245                          INC     -(R5)
002500  000000                          HALT                            ; RTS INSTRUCTION FAILED

002502  020027  002402          12$:    CMP     R0,#7$                  ; DOES R0 CONTAIN THE RETURN ADDRESS?

002506  001404                          BEQ     13$
002510  012745  000066                  MOV     #66,-(R5)
002514  005245                          INC     -(R5)
002516  000000                          HALT                            ; WRONG RETURN ADDRESS IN LINKAGE REGISTER R0
002520  012700  002422          13$:    MOV     #8$,R0                  ; SET RETURN ADDRESS AT 8$
002524  000200                          RTS     R0
```

```
002526                          ENDJSR:
002526  012745  000067                  MOV     #67,-(R5)
002532  005245                          INC     -(R5)
002534  000000                          HALT                    ; RTS INSTRUCTION FAILED OR SEQUENCE ERROR

                                ;;*******************************************************************
                                ;*TEST: 15       CHECK REGISTER SELECTION
                                ;;*******************************************************************

002536                          REGS:
002536  021527  000015                  CMP     (R5),#15
002542  001034                          BNE     EREGS           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
002544  005215                          INC     (R5)
002546  010667  175364                  MOV     R6,TEMP         ; SAVE THE STACK POINTER
002552  012700  000001                  MOV     #1,R0           ; LOAD THE REGISTERS
002556  012701  000004                  MOV     #4,R1           ;
002562  012702  000020                  MOV     #20,R2          ;
002566  012703  000100                  MOV     #100,R3         ;
002572  012704  000400                  MOV     #400,R4         ;
002576  005006                          CLR     R6              ;
002600  060006                          ADD     R0,R6           ; ADD UP THE REGISTERS
002602  060106                          ADD     R1,R6           ;
002604  060206                          ADD     R2,R6           ;
002606  060306                          ADD     R3,R6           ;
002610  060406                          ADD     R4,R6           ;
002612  060506                          ADD     R5,R6           ;
002614  022706  000627                  CMP     #$TESTN+525,R6  ; CHECK IT
002620  001003                          BNE     1$              ; FAILED
002622  016706  175310                  MOV     TEMP,R6         ; RESTORE STACK POINTER
002626  000406                          BR      TSTB0           ; CONTINUE
002630  016706  175302          1$:     MOV     TEMP,R6         ; RESTORE STACK POINTER
002634                          EREGS:
002634  012745  000070                  MOV     #70,-(R5)
002640  005245                          INC     -(R5)
002642  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
                                        ;;*******************************************************************************
002644                                  TSTB0:
002644  021527  000016                          CMP     (R5),#16
002650  001404                                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
002652  012745  000071                          MOV     #71,-(R5)
002656  005245                                  INC     -(R5)
002660  000000                                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
002662  005215                          2$:     INC     (R5)
002664  000277                                  SCC
002666  105000                                  CLRB    R0              ; CLEAR THE REGISTER
002670  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
002674  105700                                  TSTB    R0              ; CHECK IT
002676  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
002702  112701  000377                          MOVB    #377,R1         ; LOAD THE REGISTER
002706  004737  017236                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
002712  105701                                  TSTB    R1              ; CHECK IT
002714  004737  017236                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10


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

002720                                  CMPB0:
002720  021527  000017                          CMP     (R5),#17
002724  001051                                  BNE     ECMPB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
002726  005215                          1$:     INC     (R5)
002730  000277                                  SCC
002732  152702  000377                          BISB    #377,R2         ; LOAD REGISTER
002736  004737  017256                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
002742  122702  000377                          CMPB    #377,R2         ; CHECK COMPARE
002746  001404                                  BEQ     2$              ; CONTINUE IF OK
002750  012745  000072                          MOV     #72,-(R5)
002754  005245                                  INC     -(R5)
002756  000000                                  HALT                    ; BISB OR CMPB INSTRUCTION FAILED
002760  112700  000077                  2$:     MOVB    #77,R0
002764  120002                                  CMPB    R0,R2           ; CHECK IT AGAIN
002766  100004                                  BPL     3$              ; CONTINUE IF OK
002770  012745  000073                          MOV     #73,-(R5)
002774  005245                                  INC     -(R5)
002776  000000                                  HALT                    ; CMPB INSTRUCTION FAILED [WRONG CC]
003000  120200                          3$:     CMPB    R2,R0           ; ONCE MORE
003002  100404                                  BMI     4$              ; CONTINUE IF OK
003004  012745  000074                          MOV     #74,-(R5)
003010  005245                                  INC     -(R5)
003012  000000                                  HALT                    ; CMPB INSTRUCTION FAILED [WRONG CC]
003014  112702  000377                  4$:     MOVB    #377,R2         ; LOAD REGISTER, SIGN EXTEND
003020  122702  000377                          CMPB    #377,R2         ; CHECK IF BYTE INSTRUCTION
```

```
003024  001404                          BEQ     5$              ; CONTINUE IF OK
003026  012745  000075                  MOV     #75,-(R5)
003032  005245                          INC     -(R5)
003034  000000                          HALT                    ; CMPB BECAME CMP INSTRUCTION
003036  112702  000377          5$:     MOVB    #377,R2         ; LOAD REGISTER, SIGN EXTEND
003042  120227  000377                  CMPB    R2,#377         ; CHECK IF BYTE INSTRUCTION
003046  001404                          BEQ     BICB0           ; CONTINUE IF OK
003050                          ECMPB0:
003050  012745  000076                  MOV     #76,-(R5)
003054  005245                          INC     -(R5)
003056  000000                          HALT                    ; WRONG CC OR WRONG SEQUENCE
```

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

                                BICB0:
003060                                  CMP     (R5),#20
003060  021527  000020                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
003064  001404                          MOV     #77,-(R5)
003066  012745  000077                  INC     -(R5)
003072  005245                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
003074  000000                  2$:     INC     (R5)
003076  005215                          MOVB    #377,R3         ; LOAD REGISTER
003100  112703  000377                  MOVB    #252,R0         ; PLACE #252 IN R0
003104  112700  000252                  SCC
003110  000277                          BICB    R0,R3           ; CLEAR EVERY OTHER BIT
003112  140003                          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
003114  004737  017066                  BITB    R0,R3           ; CHECK IT
003120  130003                          BEQ     4$              ; CONTINUE IF OK
003122  001404                          MOV     #100,-(R5)
003124  012745  000100                  INC     -(R5)
003130  005245                          HALT                    ; BICB OR BITB INSTRUCTION FAILED
003132  000000                  4$:     BITB    #125,R3         ; CHECK IT
003134  132703  000125                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
003140  004737  017066                  BISB    R0,R3           ; SET THE BITS THAT WERE CLEARED
003144  150003                          BMI     6$
003146  100404                          MOV     #101,-(R5)
003150  012745  000101                  INC     -(R5)
003154  005245                          HALT                    ; BISB INSTRUCTION FAILED
003156  000000                  6$:     BICB    #177,R3         ; CLEAR ALL THE BITS EXCEPT FOR SIGN
003160  142703  000177                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
003164  004737  017256                  BITB    #377,R3         ; CHECK IT
003170  132703  000377                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
003174  004737  017256


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

                                INCB0:
003200                                  CMP     (R5),#21
003200  021527  000021                  BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
003204  001404                          MOV     #102,-(R5)
003206  012745  000102                  INC     -(R5)
003212  005245                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
003214  000000                  1$:     INC     (R5)
003216  005215                          MOVB    #177,R4         ; R4 = 177
003220  112704  000177                  SEC
003224  000261                          INCB    R4              ; ADD ONES INTO REG. 4
003226  105204                          JSR     PC,@#$CC13      ; CHECK FOR CC = 13
003230  004737  017320                  MOVB    #376,R4
003234  112704  000376                  INCB    R4
003240  105204                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
003242  004737  017256                  INCB    R4
003246  105204                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
003250  004737  017172                  INCB    R4
003254  105204
```

```
003256  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
003262  122704  000001                  CMPB    #1,R4           ; CHECK IT
003266  001404                          BEQ     2$              ; CONTINUE IF OK
003270  012745  000103                  MOV     #103,-(R5)
003274  005245                          INC     -(R5)
003276  000000                          HALT                    ; INCB INSTRUCTION FAILED
003300  000261                  2$:     SEC
003302  105304                          DECB    R4              ; SUBTRACT ONES FROM REG. 4
003304  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
003310  105304                          DECB    R4
003312  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
003316  012704  000200                  MOV     #200,R4
003322  105304                          DECB    R4
003324  004737  017126                  JSR     PC,@#$CC3       ; CHECK FOR CC = 3
003330  105304                          DECB    R4
003332  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
```

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

003336                          COMB0:
003336  021527  000022                  CMP     (R5),#22
003342  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
003344  012745  000104                  MOV     #104,-(R5)
003350  005245                          INC     -(R5)
003352  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
003354  005215                  1$:     INC     (R5)
003356  112703  000252                  MOVB    #252,R3         ; LOAD EVERY OTHER BIT
003362  000277                          SCC
003364  105103                          COMB    R3              ; 1'S COMPLEMENT
003366  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
003372  122703  000125                  CMPB    #125,R3         ; CHECK IT
003376  001404                          BEQ     2$              ; CONTINUE IF OK
003400  012745  000105                  MOV     #105,-(R5)
003404  005245                          INC     -(R5)
003406  000000                          HALT                    ; COMB INSTRUCTION FAILED
003410  000277                  2$:     SCC
003412  105103                          COMB    R3              ; COMPLEMENT BACK
003414  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
003420  122703  000252                  CMPB    #252,R3         ; CHECK IT
003424  001404                          BEQ     3$              ; CONTINUE IF OK
003426  012745  000106                  MOV     #106,-(R5)
003432  005245                          INC     -(R5)
003434  000000                          HALT                    ; COMB INSTRUCTION FAILED
003436  012703  000377          3$:     MOV     #377,R3
003442  000277                          SCC
003444  105103                          COMB    R3
003446  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5


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

003452                          NEGB0:
003452  021527  000023                  CMP     (R5),#23
003456  001025                          BNE     ENEGB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
003460  005215                  1$:     INC     (R5)
003462  112700  000001                  MOVB    #1,R0           ; LOAD THE REGISTER
003466  105400                          NEGB    R0              ; 2'S COMPLEMENT
003470  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
003474  122700  000377                  CMPB    #377,R0         ; CHECK IT
003500  001404                          BEQ     2$              ; CONTINUE IF OK
003502  012745  000107                  MOV     #107,-(R5)
003506  005245                          INC     -(R5)
003510  000000                          HALT                    ; NEGB INSTRUCTION FAILED
003512  012700  000200          2$:     MOV     #200,R0
003516  105400                          NEGB    R0              ; 2'S COMPLEMENT
003520  004737  017320                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
003524  122700  000200                  CMPB    #200,R0         ; CHECK IT
003530  001404                          BEQ     ROLB0           ; CONTINUE IF OK
```

```
003532                          ENEGB0:
003532  012745  000110                  MOV     #110,-(R5)
003536  005245                          INC     -(R5)
003540  000000                          HALT                    ; WRONG RESULT IN R0 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************
                                ;*TEST: 24      NEW INSTRUCTION IN THIS SECTION IS ROLB
                                ;;*******************************************************************

                                ROLB0:
003542                                  CMP     (R5),#24
003542  021527  000024                  BNE     EROLB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
003546  001026                          INC     (R5)
003550  005215                          MOVB    #40,R1          ; LOAD REGISTER
003552  112701  000040                  CCC                     ; CLEAR FLAGS
003556  000257                          ROLB    R1              ; SHIFT
003560  106101                          ROLB    R1              ;
003562  106101                          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
003564  004737  017300                  CMPB    #200,R1         ; CHECK IT
003570  122701  000200                  BEQ     1$              ; CONTINUE IF OK
003574  001404                          MOV     #111,-(R5)
003576  012745  000111                  INC     -(R5)
003602  005245                          HALT                    ; ROLB INSTRUCTION FAILED
003604  000000                  1$:     ROLB    R1              ; SHIFT
003606  106101                          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
003610  004737  017214                  ROLB    R1              ; SHIFT
003614  106101                          CMPB    #1,R1           ; CHECK IT
003616  122701  000001                  BEQ     RORB0           ; CONTINUE IF OK
003622  001404                  EROLB0:
003624                                  MOV     #112,-(R5)
003624  012745  000112                  INC     -(R5)
003630  005245                          HALT                    ; WRONG RESULT IN R1 OR WRONG SEQUENCE
003632  000000


                                ;;*******************************************************************
                                ;*TEST: 25      NEW INSTRUCTION IN THIS SECTION IS RORB
                                ;;*******************************************************************

                                RORB0:
003634                                  CMP     (R5),#25
003634  021527  000025                  BNE     ERORB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
003640  001026                          INC     (R5)
003642  005215                          MOVB    #4,R2           ; LOAD REGISTER
003644  112702  000004                  CCC                     ; CLEAR FLAGS
003650  000257                          RORB    R2              ; SHIFT
003652  106002                          RORB    R2              ;
003654  106002                          CMPB    #1,R2           ; CHECK IT
003656  122702  000001                  BEQ     1$              ; CONTINUE IF OK
003662  001404                          MOV     #113,-(R5)
003664  012745  000113                  INC     -(R5)
003670  005245                          HALT                    ; RORB INSTRUCTION FAILED
003672  000000                  1$:     RORB    R2              ; SHIFT
003674  106002                          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
003676  004737  017214                  RORB    R2              ; SHIFT
003702  106002                          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
003704  004737  017300                  CMPB    #200,R2         ; CHECK IT
003710  122702  000200                  BEQ     ASLB0           ; CONTINUE IF OK
003714  001404                  ERORB0:
003716                                  MOV     #114,-(R5)
003716  012745  000114                  INC     -(R5)
003722  005245                          HALT
003724  000000
```

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

003726                          ASLB0:
003726  021527  000026                  CMP     (R5),#26
003732  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
003734  012745  000115                  MOV     #115,-(R5)
003740  005245                          INC     -(R5)
003742  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
003744  005215                  2$:     INC     (R5)
003746  112703  000040                  MOVB    #40,R3          ; LOAD REGISTER
003752  000257                          CCC                     ; CLEAR FLAGS
003754  106303                          ASLB    R3              ; SHIFT
003756  106303                          ASLB    R3              ;
003760  004737  017300                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
003764  122703  000200                  CMPB    #200,R3         ; CHECK IT
003770  001404                          BEQ     4$              ; CONTINUE IF OK
003772  012745  000116                  MOV     #116,-(R5)
003776  005245                          INC     -(R5)
004000  000000                          HALT                    ; ASLB INSTRUCTION FAILED
004002  106303                  4$:     ASLB    R3              ; SHIFT
004004  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
004010  106303                          ASLB    R3              ; SHIFT
004012  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4

                                ;;*******************************************************************
                                ;*TEST: 27      NEW INSTRUCTION IN THIS SECTION IS ASRB
                                ;;*******************************************************************

004016                          ASRB0:
004016  021527  000027                  CMP     (R5),#27
004022  001034                          BNE     EASRB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
004024  005215                  1$:     INC     (R5)
004026  112704  000004                  MOVB    #4,R4           ; LOAD REGISTER
004032  000257                          CCC                     ; CLEAR FLAGS
004034  106204                          ASRB    R4              ; SHIFT
004036  106204                          ASRB    R4              ;
004040  122704  000001                  CMPB    #1,R4           ; CHECK IT
004044  001404                          BEQ     2$              ; CONTINUE IF OK
004046  012745  000117                  MOV     #117,-(R5)
004052  005245                          INC     -(R5)
004054  000000                          HALT                    ; ASRB INSTRUCTION FAILED
004056  106204                  2$:     ASRB    R4              ; SHIFT
004060  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
004064  106204                          ASRB    R4              ; SHIFT
004066  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
004072  112703  000202                  MOVB    #202,R3         ; LOAD REGISTER
004076  106203                          ASRB    R3              ; SHIFT
004100  106203                          ASRB    R3              ;
004102  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
004106  122703  000340                  CMPB    #340,R3         ; CHECK IT
004112  001404                          BEQ     ADCB0           ; CONTINUE IF OK
004114                          EASRB0:
004114  012745  000120                  MOV     #120,-(R5)
004120  005245                          INC     -(R5)
```

```
1297  004122  000000                    HALT
```

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

004124                          ADCB0:
004124  021527  000030                  CMP     (R5),#30
004130  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
004132  012745  000121                  MOV     #121,-(R5)
004136  005245                          INC     -(R5)
004140  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
004142  005215                  2$:     INC     (R5)
004144  105000                          CLRB    R0              ; CLEAR THE REGISTER
004146  000257                          CCC                     ; CLEAR FLAGS
004150  105500                          ADCB    R0              ; ADD C BIT = 0
004152  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
004156  000261                          SEC                     ; C=1
004160  105500                          ADCB    R0              ; ADD C BIT=1
004162  000261                          SEC                     ; C=1
004164  105500                          ADCB    R0              ; AGAIN
004166  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
004172  122700  000002                  CMPB    #2,R0           ; CHECK IT
004176  001404                          BEQ     4$              ; CONTINUE IF OK
004200  012745  000122                  MOV     #122,-(R5)
004204  005245                          INC     -(R5)
004206  000000                          HALT                    ; ADCB INSTRUCTION FAILED
004210  112700  000177          4$:     MOVB    #177,R0         ; LOAD LARGEST POSITVE NUMBER
004214  000261                          SEC                     ; C=1
004216  105500                          ADCB    R0              ; ADD C BIT=1
004220  004737  017300                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
004224  122700  000200                  CMPB    #200,R0         ; CHECK IT
004230  001404                          BEQ     6$              ; CONTINUE IF OK
004232  012745  000123                  MOV     #123,-(R5)
004236  005245                          INC     -(R5)
004240  000000                          HALT                    ; ADCB INSTRUCTION FAILED
004242  112700  000377          6$:     MOVB    #377,R0         ; LOAD -1
004246  000261                          SEC                     ; C=1
004250  105500                          ADCB    R0              ; ADD C BIT=1
004252  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5


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

004256                          SBCB0:
004256  021527  000031                  CMP     (R5),#31
004262  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
004264  012745  000124                  MOV     #124,-(R5)
004270  005245                          INC     -(R5)
004272  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
004274  005215                  1$:     INC     (R5)            ; TEST IS IN WRONG SEQUENCE
004276  112701  000003                  MOVB    #3,R1           ; LOAD REGISTER
004302  000257                          CCC                     ; CLEAR FLAGS
004304  105601                          SBCB    R1              ; SUBTRACT C BIT=0
004306  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
```

```
004312  122701  000003          CMPB    #3,R1           ; CHECK IT
004316  001404                  BEQ     2$              ; CONTINUE IF OK
004320  012745  000125          MOV     #125,-(R5)
004324  005245                  INC     -(R5)
004326  000000                  HALT                    ; SBCB INSTRUCTION FAILED
004330  000261          2$:     SEC                     ; C=1
004332  105601                  SBCB    R1              ; SUBTRACT C BIT=1
004334  000261                  SEC                     ; C=1
004336  105601                  SBCB    R1              ;
004340  004737  017046          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
004344  122701  000001          CMPB    #1,R1           ; CHECK IT
004350  001404                  BEQ     3$              ; CONTINUE IF OK
004352  012745  000126          MOV     #126,-(R5)
004356  005245                  INC     -(R5)
004360  000000                  HALT                    ; SBCB INSTRUCTION FAILED
004362  000261          3$:     SEC                     ; C=1
004364  105601                  SBCB    R1              ; SUBTRACT C BIT=1
004366  004737  017150          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
004372  000261                  SEC                     ; C=1
004374  105601                  SBCB    R1              ; SUBTRACT C BIT = 1
004376  004737  017256          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
004402  122701  000377          CMPB    #377,R1         ; CHECK IT
004406  001404                  BEQ     4$              ; CONTINUE IF OK
004410  012745  000127          MOV     #127,-(R5)
004414  005245                  INC     -(R5)
004416  000000                  HALT                    ; SBCB INSTRUCTION FAILED
004420  112701  000200          MOVB    #200,R1         ; LOAD R1
004424  000261          4$:     SEC                     ; C=1
004426  105601                  SBCB    R1              ; SUBTRACT C BIT = 1
004430  004737  017106          JSR     PC,@#$CC2       ; CHECK FOR CC = 2
```

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


                                        ;;*******************************************************************************
                                        ;*TEST: 32       NEW INSTRUCTIONS USED IN THIS SECTION ARE TST, CLR, MOV
                                        ;;*******************************************************************************

004434                                  TST0:
004434  021527  000032                          CMP     (R5),#32
004440  001404                                  BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
004442  012745  000130                          MOV     #130,-(R5)
004446  005245                                  INC     -(R5)
004450  000000                                  HALT                            ; TEST IS IN WRONG SEQUENCE
004452  005215                          1$:     INC     (R5)
004454  000277                                  SCC
004456  005000                                  CLR     R0                      ; CLEAR THE REGISTER
004460  004737  017150                          JSR     PC,@#$CC4               ; CHECK FOR CC = 4
004464  005700                                  TST     R0                      ; CHECK IT
004466  004737  017150                          JSR     PC,@#$CC4               ; CHECK FOR CC = 4
004472  012704  177777                          MOV     #177777,R4              ; LOAD THE REGISTER
004476  010401                                  MOV     R4,R1
004500  004737  017236                          JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004504  005701                                  TST     R1                      ; CHECK IT
004506  004737  017236                          JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004512  020401                                  CMP     R4,R1                   ; CHECK R1 TO CONTAIN PROPER DATA
004514  001404                                  BEQ     2$
004516  012745  000131                          MOV     #131,-(R5)
004522  005245                                  INC     -(R5)
004524  000000                                  HALT
004526  000263                          2$:     SEVC                            ; SET V & C BITS
004530  010000                                  MOV     R0,R0
004532  004767  012434                          JSR     PC,$CC5


                                        ;;*******************************************************************************
                                        ;*TEST: 33       NEW INSTRUCTIONS USED IN THIS SECTION ARE CMP, BIS
                                        ;;*******************************************************************************

004536                                  CMP0:
004536  021527  000033                          CMP     (R5),#33
004542  001026                                  BNE     ECMP0                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
004544  005215                          1$:     INC     (R5)
004546  012700  177777                          MOV     #177777,R0              ; LOAD REGISTER
004552  050002                                  BIS     R0,R2                   ; CHECK THE BIS INSTRUCTION
004554  004737  017236                          JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004560  020002                                  CMP     R0,R2                   ; CHECK COMPARE
004562  001404                                  BEQ     2$                      ; CONTINUE IF OK
004564  012745  000132                          MOV     #132,-(R5)
004570  005245                                  INC     -(R5)
004572  000000                                  HALT                            ; BIS OR CMP INSTRUCTION FAILED
```

```
004574  022702  000077        2$:     CMP     #77,R2          ; CHECK IT AGAIN
004600  100004                        BPL     3$              ; CONTINUE IF OK
004602  012745  000133                MOV     #133,-(R5)
004606  005245                        INC     -(R5)
004610  000000                        HALT                    ; CMP INSTRUCTION FAILED [WRONG CC]
004612  020227  000077        3$:     CMP     R2,#77          ; ONCE MORE
004616  100404                        BMI     BICO            ; CONTINUE IF OK
004620                        ECMPO:
004620  012745  000134                MOV     #134,-(R5)
004624  005245                        INC     -(R5)
004626  000000                        HALT                    ; WRONG CC OR WRONG SEQUENCE
```

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

004630                          BIC0:
004630  021527  000034                  CMP     (R5),#34
004634  001053                          BNE     EBIC0           ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
004636  005215                          INC     (R5)
004640  012703  177777                  MOV     #177777,R3      ; LOAD REGISTER
004644  012700  000136                  MOV     #TEMP,R0        ; PLACE THE ADDRESS OF LOCATION TEMP IN R0
004650  012710  125252                  MOV     #125252,(R0)    ; SET (R0)
004654  000277                          SCC
004656  041003                          BIC     (R0),R3         ; CLEAR EVERY OTHER BIT
004660  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
004664  031003                          BIT     (R0),R3         ; CHECK IT
004666  001404                          BEQ     1$              ; CONTINUE IF OK
004670  012745  000135                  MOV     #135,-(R5)
004674  005245                          INC     -(R5)
004676  000000                          HALT                    ; BIC OR BIT INSTRUCTION FAILED
004700  032703  052525          1$:     BIT     #52525,R3       ; CHECK IT
004704  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
004710  052703  125252                  BIS     #125252,R3      ; SET THE BITS THAT WERE CLEARED
004714  100404                          BMI     2$              ; CONTINUE IF OK
004716  012745  000136                  MOV     #136,-(R5)
004722  005245                          INC     -(R5)
004724  000000                          HALT                    ; BIT OR BIS INSTRUCTION FAILED
004726  042703  077777          2$:     BIC     #77777,R3       ; CLEAR ALL THE BITS EXCEPT FOR SIGN
004732  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
004736  012700  177777                  MOV     #177777,R0
004742  030003                          BIT     R0,R3           ; CHECK IT
004744  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
004750  000263                          SEVC                    ; SET V & C BITS
004752  040000                          BIC     R0,R0
004754  004737  017172                  JSR     PC,@#$CC5       ; CHECK CC = 5
004760  005700                          TST     R0              ; CHECK R0 TO CONTAIN 0
004762  001404                          BEQ     INC0
004764                          EBIC0:
004764  012745  000137                  MOV     #137,-(R5)
004770  005245                          INC     -(R5)
004772  000000                          HALT                    ; BIC FAILED OR SEQUENCE ERROR
```

1492

```
                                        ;;*******************************************************************************
                                        ;*TEST: 35      NEW INSTRUCTIONS USED IN THIS SECTION ARE INC, DEC
                                        ;;*******************************************************************************

004774                                  INC0:
004774  021527  000035                          CMP     (R5),#35
005000  001404                                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
005002  012745  000140                          MOV     #140,-(R5)
005006  005245                                  INC     -(R5)
005010  000000                                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
005012  005215                          2$:     INC     (R5)
005014  012704  077777                          MOV     #77777,R4       ; R4=77777
```

```
1506  005020  000261                    SEC
1507  005022  005204                    INC     R4              ; ADD ONES INTO REG. 4
1508  005024  004737  017320           JSR     PC,@#$CC13      ; CHECK FOR CC = 13
1509  005030  012704  177776           MOV     #177776,R4
1510  005034  005204                    INC     R4
1511  005036  004737  017256           JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1512  005042  005204                    INC     R4
1513  005044  004737  017172           JSR     PC,@#$CC5       ; CHECK FOR CC = 5
1514  005050  005204                    INC     R4
1515  005052  004737  017066           JSR     PC,@#$CC1       ; CHECK FOR CC = 1
1516  005056  022704  000001           CMP     #1,R4           ; CHECK IT
1517  005062  001404                    BEQ     4$              ; FAILED
1518  005064  012745  000141           MOV     #141,-(R5)
1519  005070  005245                    INC     -(R5)
1520  005072  000000                    HALT                    ; INC INSTRUCTION FAILED
1521  005074  000261            4$:     SEC
1522  005076  005304                    DEC     R4              ; SUBTRACT ONES FROM REG. 4
1523  005100  004737  017172           JSR     PC,@#$CC5       ; CHECK FOR CC = 5
1524  005104  005304                    DEC     R4
1525  005106  004737  017256           JSR     PC,@#$CC11      ; CHECK FOR CC = 11
1526  005112  012704  100000           MOV     #100000,R4
1527  005116  005304                    DEC     R4              ;
1528  005120  004737  017126           JSR     PC,@#$CC3       ; CHECK FOR CC = 3
1529  005124  005304                    DEC     R4              ;
1530  005126  004737  017066           JSR     PC,@#$CC1       ; CHECK FOR CC = 1
```

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

005132                          COM0:
005132  021527  000036                  CMP     (R5),#36
005136  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
005140  012745  000142                  MOV     #142,-(R5)
005144  005245                          INC     -(R5)
005146  000000                          HALT                    ; TEST IS IN WRONG SEQUENCE
005150  005215                  1$:     INC     (R5)
005152  012703  125252                  MOV     #125252,R3      ; LOAD EVERY OTHER BIT
005156  000277                          SCC
005160  005103                          COM     R3              ; 1'S COMPLEMENT
005162  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
005166  022703  052525                  CMP     #52525,R3       ; CHECK IT
005172  001404                          BEQ     2$              ; CONTINUE IF OK
005174  012745  000143                  MOV     #143,-(R5)
005200  005245                          INC     -(R5)
005202  000000                          HALT                    ; COM INSTRUCTION FAILED
005204  000277                  2$:     SCC
005206  005103                          COM     R3              ; COMPLEMENT BACK
005210  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
005214  022703  125252                  CMP     #125252,R3      ; CHECK IT
005220  001404                          BEQ     3$              ; CONTINUE IF OK
005222  012745  000144                  MOV     #144,-(R5)
005226  005245                          INC     -(R5)
005230  000000                          HALT                    ; COM INSTRUCTION FAILED
005232  012703  177777          3$:     MOV     #177777,R3
005236  000277                          SCC
005240  005103                          COM     R3
005242  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5


                                ;;*******************************************************************************
                                ;;*TEST: 37       NEW INSTRUCTION IN THIS SECTION IS NEG
                                ;;*******************************************************************************

005246                          NEG0:
005246  021527  000037                  CMP     (R5),#37
005252  001025                          BNE     ENEG0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
005254  005215                  1$:     INC     (R5)
005256  012700  000001                  MOV     #1,R0           ; LOAD THE REGISTER
005262  005400                          NEG     R0              ; 2'S COMPLEMENT
005264  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
005270  022700  177777                  CMP     #177777,R0      ; CHECK IT
005274  001404                          BEQ     2$              ; CONTINUE IF OK
005276  012745  000145                  MOV     #145,-(R5)
005302  005245                          INC     -(R5)
005304  000000                          HALT                    ; NEG INSTRUCTION FAILED
005306  012700  100000          2$:     MOV     #100000,R0
005312  005400                          NEG     R0              ; 2'S COMPLEMENT
005314  004737  017320                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
005320  022700  100000                  CMP     #100000,R0      ; CHECK IT
005324  001404                          BEQ     ROL0            ; CONTINUE IF OK
```

```
1587  005326                          ENEGO:
1588  005326  012745  000146                 MOV     #146,-(R5)
1589  005332  005245                         INC     -(R5)
1590  005334  000000                         HALT                    ; WRONG RESULT IN R0 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************
                                ;*TEST: 40      NEW INSTRUCTION IN THIS SECTION IS ROL
                                ;;*******************************************************************

005336                          ROL0:
005336  021527  000040                  CMP     (R5),#40
005342  001026                          BNE     EROL0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
005344  005215                          INC     (R5)
005346  012701  020000                  MOV     #20000,R1       ; LOAD REGISTER
005352  000257                          CCC                     ; CLEAR FLAGS
005354  006101                          ROL     R1              ; SHIFT
005356  006101                          ROL     R1              ;
005360  004737  017300                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
005364  022701  100000                  CMP     #100000,R1      ; CHECK IT
005370  001404                          BEQ     1$              ; CONTINUE IF OK
005372  012745  000147                  MOV     #147,-(R5)
005376  005245                          INC     -(R5)
005400  000000                          HALT                    ; ROL INSTRUCTION FAILED
005402  006101                  1$:     ROL     R1              ; SHIFT
005404  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
005410  006101                          ROL     R1              ; SHIFT
005412  022701  000001                  CMP     #1,R1           ; CHECK IT
005416  001404                          BEQ     ROR0            ; CONTINUE IF OK
005420                          EROL0:
005420  012745  000150                  MOV     #150,-(R5)
005424  005245                          INC     -(R5)
005426  000000                          HALT                    ; WRONG RESULT IN R1 OR WRONG SEQUENCE


                                ;;*******************************************************************
                                ;*TEST: 41      NEW INSTRUCTION IN THIS SECTION IS ROR
                                ;;*******************************************************************

005430                          ROR0:
005430  021527  000041                  CMP     (R5),#41
005434  001026                          BNE     EROR0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
005436  005215                          INC     (R5)
005440  012702  000004                  MOV     #4,R2           ; LOAD REGISTER
005444  000257                          CCC                     ; CLEAR FLAGS
005446  006002                          ROR     R2              ; SHIFT
005450  006002                          ROR     R2              ;
005452  022702  000001                  CMP     #1,R2           ; CHECK IT
005456  001404                          BEQ     1$              ; CONTINUE IF OK
005460  012745  000151                  MOV     #151,-(R5)
005464  005245                          INC     -(R5)
005466  000000                          HALT                    ; ROR INSTRUCTION FAILED
005470  006002                  1$:     ROR     R2              ; SHIFT
005472  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
005476  006002                          ROR     R2              ; SHIFT
005500  004737  017300                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
005504  022702  100000                  CMP     #100000,R2      ; CHECK IT
005510  001404                          BEQ     ASL0            ; CONTINUE IF OK
005512                          EROR0:
005512  012745  000152                  MOV     #152,-(R5)
005516  005245                          INC     -(R5)
```

```
1647  005520  000000                    HALT                    ; WRONG RESULT IN R2 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;*TEST: 42      NEW INSTRUCTION IN THIS SECTION IS ASL
                                ;;*******************************************************************************

005522                          ASL0:
005522  021527  000042                  CMP     (R5),#42
005526  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
005530  012745  000153                  MOV     #153,-(R5)
005534  005245                          INC     -(R5)
005536  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
005540  005215                  2$:     INC     (R5)
005542  012703  020000                  MOV     #20000,R3       ; LOAD REGISTER
005546  000257                          CCC                     ; CLEAR FLAGS
005550  006303                          ASL     R3              ; SHIFT
005552  006303                          ASL     R3              ;
005554  004737  017300                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
005560  022703  100000                  CMP     #100000,R3      ; CHECK IT
005564  001404                          BEQ     4$              ; CONTINUE IF OK
005566  012745  000154                  MOV     #154,-(R5)
005572  005245                          INC     -(R5)
005574  000000                          HALT                    ; ASL INSTRUCTION FAILED
005576  006303                  4$:     ASL     R3              ; SHIFT
005600  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
005604  006303                          ASL     R3              ; SHIFT
005606  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4

                                ;;*******************************************************************************
                                ;*TEST: 43      NEW INSTRUCTION IN THIS SECTION IS ASR
                                ;;*******************************************************************************

005612                          ASR0:
005612  021527  000043                  CMP     (R5),#43
005616  001034                          BNE     EASR0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
005620  005215                  1$:     INC     (R5)
005622  012704  000004                  MOV     #4,R4           ; LOAD REGISTER
005626  000257                          CCC                     ; CLEAR FLAGS
005630  006204                          ASR     R4              ; SHIFT
005632  006204                          ASR     R4              ;
005634  022704  000001                  CMP     #1,R4           ; CHECK IT
005640  001404                          BEQ     2$              ; CONTINUE IF OK
005642  012745  000155                  MOV     #155,-(R5)
005646  005245                          INC     -(R5)
005650  000000                          HALT                    ; ASR INSTRUCTION FAILED
005652  006204                  2$:     ASR     R4              ; SHIFT
005654  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
005660  006204                          ASR     R4              ; SHIFT
005662  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
005666  012703  100002                  MOV     #100002,R3      ; LOAD REGISTER
005672  006203                          ASR     R3              ; SHIFT
005674  006203                          ASR     R3              ;
005676  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
005702  022703  160000                  CMP     #160000,R3      ; CHECK IT
005706  001404                          BEQ     ADC0            ; CONTINUE IF OK
005710                          EASR0:
005710  012745  000156                  MOV     #156,-(R5)
005714  005245                          INC     -(R5)
```

```
1704  005716  000000                    HALT                    ; WRONG RESULT IN R3 OR WRONG SEQUENCE
```

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

005720                                  ADC0:
005720  021527  000044                          CMP     (R5),#44
005724  001404                                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
005726  012745  000157                          MOV     #157,-(R5)
005732  005245                                  INC     -(R5)
005734  000000                                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
005736  005215                          2$:     INC     (R5)
005740  005000                                  CLR     R0              ; CLEAR THE REGISTER
005742  000257                                  CCC                     ; CLEAR FLAGS
005744  005500                                  ADC     R0              ; ADD C BIT = 0
005746  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
005752  000261                                  SEC                     ; C=1
005754  005500                                  ADC     R0              ; ADD C BIT=1
005756  000261                                  SEC                     ; C=1
005760  005500                                  ADC     R0              ; AGAIN
005762  004737  017046                          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
005766  022700  000002                          CMP     #2,R0           ; CHECK IT
005772  001404                                  BEQ     4$              ; CONTINUE IF OK
005774  012745  000160                          MOV     #160,-(R5)
006000  005245                                  INC     -(R5)
006002  000000                                  HALT                    ; ADC INSTRUCTION FAILED
006004  012700  077777                  4$:     MOV     #77777,R0       ; LOAD LARGEST POSITVE NUMBER
006010  000261                                  SEC                     ; C=1
006012  005500                                  ADC     R0              ; ADD C BIT=1
006014  004737  017300                          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
006020  022700  100000                          CMP     #100000,R0      ; CHECK IT
006024  001404                                  BEQ     6$              ; FAILED
006026  012745  000161                          MOV     #161,-(R5)
006032  005245                                  INC     -(R5)
006034  000000                                  HALT                    ; ADC INSTRUCTION FAILED
006036  012700  177777                  6$:     MOV     #-1,R0          ; LOAD -1
006042  000261                                  SEC                     ; C=1
006044  005500                                  ADC     R0              ; ADD C BIT=1
006046  004737  017172                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5


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

006052                                  SBC0:
006052  021527  000045                          CMP     (R5),#45
006056  001404                                  BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
006060  012745  000162                          MOV     #162,-(R5)
006064  005245                                  INC     -(R5)
006066  000000                                  HALT                    ; TEST IS IN WRONG SEQUENCE
006070  005215                          1$:     INC     (R5)
006072  012701  000003                          MOV     #3,R1           ; LOAD REGISTER
006076  000257                                  CCC                     ; CLEAR FLAGS
006100  005601                                  SBC     R1              ; SUBTRACT C BIT=0
006102  004737  017046                          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
```

```
006106  022701  000003                  CMP     #3,R1           ; CHECK IT
006112  001404                          BEQ     2$              ; CONTINUE IF OK
006114  012745  000163                  MOV     #163,-(R5)
006120  005245                          INC     -(R5)
006122  000000                          HALT                    ; SBC INSTRUCTION FAILED
006124  000261                  2$:     SEC                     ; C=1
006126  005601                          SBC     R1              ; SUBTRACT C BIT=1
006130  000261                          SEC                     ; C=1
006132  005601                          SBC     R1              ;
006134  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
006140  022701  000001                  CMP     #1,R1           ; CHECK IT
006144  001404                          BEQ     3$              ; CONTINUE IF OK
006146  012745  000164                  MOV     #164,-(R5)
006152  005245                          INC     -(R5)
006154  000000                          HALT                    ; SBC INSTRUCTION FAILED
006156  000261                  3$:     SEC                     ; C=1
006160  005601                          SBC     R1              ; SUBTRACT C BIT=1
006162  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
006166  000261                          SEC                     ; C=1
006170  005601                          SBC     R1              ; SUBTRACT C BIT = 1
006172  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
006176  022701  177777                  CMP     #-1,R1          ; CHECK IT
006202  001404                          BEQ     4$              ; CONTINUE IF F OK
006204  012745  000165                  MOV     #165,-(R5)
006210  005245                          INC     -(R5)
006212  000000                          HALT                    ; SBC INSTRUCTION FAILED
006214  012701  100000          4$:     MOV     #100000,R1      ; LOAD R1
006220  000261                          SEC                     ; C=1
006222  005601                          SBC     R1              ; SUBTRACT C BIT = 1
006224  004737  017106                  JSR     PC,@#$CC2       ; CHECK FOR CC = 2
```

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

006230                          SXT0:
006230  021527  000046                  CMP     (R5),#46
006234  001024                          BNE     ESXT0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
006236  005215                  1$:     INC     (R5)
006240  005002                          CLR     R2              ; CLEAR REGISTER
006242  000277                          SCC
006244  000254                          CLNZ
006246  006702                          SXT     R2              ; SIGN EXTEND
006250  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
006254  005702                          TST     R2              ; REG. 2 SHOULD STILL BE 0
006256  001404                          BEQ     2$              ; CONTINUE IF OK
006260  012745  000166                  MOV     #166,-(R5)
006264  005245                          INC     -(R5)
006266  000000                          HALT                    ; SXT INSTRUCTION FAILED
006270  000273                  2$:     SENVC                   ; SET N, V & C BITS
006272  006702                          SXT     R2              ; SIGN EXTEND
006274  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
006300  022702  177777                  CMP     #-1,R2          ; REG. 2 SHOULD NOW HAVE -1
006304  001404                          BEQ     SWAB0           ; CONTINUE IF OK
006306                          ESXT0:
006306  012745  000167                  MOV     #167,-(R5)
006312  005245                          INC     -(R5)
006314  000000                          HALT                    ; WRONG RESULT IN R2 OR WRONG SEQUENCE


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

006316                          SWAB0:
006316  021527  000047                  CMP     (R5),#47
006322  001031                          BNE     ESWAB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
006324  005215                          INC     (R5)
006326  012703  125125                  MOV     #125125,R3      ; LOAD BIT PATTERN INTO REGISTER
006332  000277                          SCC
006334  000250                          CLN
006336  000303                          SWAB    R3              ; SWAP BYTES OF REGISTER
006340  004737  017236                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
006344  022703  052652                  CMP     #52652,R3       ; CHECK IT
006350  001404                          BEQ     1$              ; CONTINUE IF OK
006352  012745  000170                  MOV     #170,-(R5)
006356  005245                          INC     -(R5)
006360  000000                          HALT                    ; SWAB INSTRUCTION FAILED
006362  012703  000377          1$:     MOV     #377,R3
006366  000277                          SCC
006370  000244                          CLZ
006372  000303                          SWAB    R3
006374  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
006400  022703  177400                  CMP     #177400,R3
006404  001404                          BEQ     XCB0
006406                          ESWAB0:
```

```
1847  006406  012745  000171          MOV     #171,-(R5)
1848  006412  005245                  INC     -(R5)
1849  006414  000000                  HALT                    ; WRONG RESULT IN R3 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;*TEST: 50      NEW INSTRUCTION IN THIS SECTION IS XOR
                                ;;*******************************************************************************

006416                          XOR0:
006416  021527  000050                  CMP     (R5),#50
006422  001034                          BNE     EXOR0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
006424  005215                          INC     (R5)
006426  012704  177777                  MOV     #-1,R4          ; LOAD REGISTERS
006432  012703  177777                  MOV     #-1,R3          ;
006436  000277                          SCC
006440  074403                          XOR     R4,R3           ; SHOULD PRODUCE 0'S IN REG. 3
006442  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
006446  012703  077777                  MOV     #77777,R3
006452  010400                          MOV     R4,R0           ; PLACE A -1 IN R0
006454  000263                          SEVC                    ; SET V & C BITS
006456  000244                          CLZ
006460  074003                          XOR     R0,R3
006462  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
006466  012702  125252                  MOV     #125252,R2      ; LOAD REGISTERS
006472  012704  052525                  MOV     #52525,R4       ;
006476  000277                          SCC
006500  074204                          XOR     R2,R4           ; SHOULD PRODUCE ALL 1'S IN REG. 4
006502  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
006506  022704  177777                  CMP     #-1,R4          ; CHECK IT
006512  001404                          BEQ     ADD0            ; CONTINUE IF OK
006514                          EXOR0:
006514  012745  000172                  MOV     #172,-(R5)
006520  005245                          INC     -(R5)
006522  000000                          HALT                    ; WRONG RESULT IN R4 OR WRONG SEQUENCE


                                ;;*******************************************************************************
                                ;*TEST: 51      NEW INSTRUCTION IN THIS SECTION IS ADD
                                ;;*******************************************************************************

006524                          ADD0:
006524  021527  000051                  CMP     (R5),#51
006530  001055                          BNE     EADD0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
006532  005215                          INC     (R5)
006534  012701  021421                  MOV     #21421,R1       ; LOAD REGISTERS
006540  060101                          ADD     R1,R1           ; ADD
006542  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
006546  022701  043042                  CMP     #43042,R1       ; CHECK IT
006552  001404                          BEQ     1$              ; CONTINUE IF OK
006554  012745  000173                  MOV     #173,-(R5)
006560  005245                          INC     -(R5)
006562  000000                          HALT                    ; ADD INSTRUCTION FAILED
006564  012700  156357          1$:     MOV     #-21421,R0      ; LOAD REGISTERS
006570  060000                          ADD     R0,R0           ; ADD
006572  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
006576  022700  134736                  CMP     #-43042,R0      ; CHECK IT
006602  001404                          BEQ     2$              ; CONTINUE IF OK
006604  012745  000174                  MOV     #174,-(R5)
006610  005245                          INC     -(R5)
```

```
006612  000000                          HALT                    ; ADD INSTRUCTION FAILED
006614  012702  100000          2$:     MOV     #100000,R2      ; LOAD REGISTERS
006620  060202                          ADD     R2,R2           ; ADD SHOULD RESULT AS 0'S
006622  004737  017214                  JSR     PC,@#$CC7       ;CHECK FOR CC = 7
006626  012704  021421                  MOV     #21421,R4       ; LOAD REGISTERS
006632  012701  156357                  MOV     #-21421,R1      ;
006636  060401                          ADD     R4,R1           ; ADD SHOULD RESULT AS 0'S
006640  001404                          BEQ     3$              ; CONTINUE IF OK
006642  012745  000175                  MOV     #175,-(R5)
006646  005245                          INC     -(R5)
006650  000000                          HALT                    ; ADD INSTRUCTION FAILED
006652  005404                  3$:     NEG     R4              ; SWITCH SOURCE AND DESTINATION
006654  012701  021421                  MOV     #21421,R1       ;
006660  060104                          ADD     R1,R4           ; SHOULD RESULT AS 0'S
006662  001404                          BEQ     SUB0            ; CONTINUE IF OK
006664                          FADD0:
006664  012745  000176                  MOV     #176,-(R5)
006670  005245                          INC     -(R5)
006672  000000                          HALT                    ; WRONG RESULT IN R1 OR WRONG SEQUENCE
```

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

006674                          SUB0:
006674  021527  000052                  CMP     (R5),#52
006700  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
006702  012745  000177                  MOV     #177,-(R5)
006706  005245                          INC     -(R5)
006710  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
006712  005215                  2$:     INC     (R5)
006714  012702  021421                  MOV     #21421,R2       ; LOAD REGISTERS
006720  012703  156357                  MOV     #-21421,R3      ;
006724  160203                          SUB     R2,R3           ; RESULT SHOULD=-43042
006726  004737  017236                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
006732  022703  134736                  CMP     #-43042,R3      ; CHECK IT
006736  001404                          BEQ     4$              ; CONTINUE IF OK
006740  012745  000200                  MOV     #200,-(R5)
006744  005245                          INC     -(R5)
006746  000000                          HALT                    ; SUB INSTRUCTION FAILED
006750  012703  021421          4$:     MOV     #21421,R3       ; LOAD REGISTER
006754  010204                          MOV     R2,R4           ; NOW R4 = #21421
006756  160403                          SUB     R4,R3           ; RESULT SHOULD=0
006760  001404                          BEQ     6$
006762  012745  000201                  MOV     #201,-(R5)
006766  005245                          INC     -(R5)
006770  000000                          HALT                    ; SUB NSTRUCTION FAILED
006772  012703  177777          6$:     MOV     #-1,R3          ; LOAD REGISTERS
006776  012702  077777                  MOV     #77777,R2       ; LOAD REGISTERS
007002  160302                          SUB     R3,R2           ; RESULT SHOULD BE 100000 AND OVERFLOW
007004  004737  017320                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
007010  022702  100000                  CMP     #100000,R2      ; CHECK IT
007014  001404                          BEQ     8$              ; CONTINUE IF OK
007016  012745  000202                  MOV     #202,-(R5)
007022  005245                          INC     -(R5)
007024  000000                          HALT                    ; SUB INSTRUCTION FAILED
007026  012704  177777          8$:     MOV     #-1,R4
007032  160304                          SUB     R3,R4
007034  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4


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

007040                          PSW:
007040  021527  000053                  CMP     (R5),#53
007044  001032                          BNE     EPSW            ; IF IN WRONG SEQUENCE THEN GO TO HLT AT THE END OF THE
007046  005215                  1$:     INC     (R5)
007050  012701  177777                  MOV     #177777,R1
007054  005000                          CLR     R0
007056                                  MTPS    R0              ; SET PSW TO 0
007056  106400                          .WORD   106400!..C
007060  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
007064                                  MFPS    R1              ; MOVE PSW TO R1
007064  106701                          .WORD   106700!..C
```

```
007066  001404                          BEQ     2$              ; CONTINUE IF BIT 8 OF PSW WAS EXTENDED IN R1
007070  012745  000203                  MOV     #203,-(R5)
007074  005245                          INC     -(R5)
007076  000000                          HALT                    ; MTPS OR MFPS INSTRUCTION FAILED
007100  004737  017150          2$:     JSR     PC,@#$CC4       ; CHECK FOR CC = 4
007104  012700  000377                  MOV     #377,R0
007110                                  MTPS    R0              ; SET PSW TO 357 SINCE MTPS DOES NOT SET T BIT
007110  106400                          .WORD   106400!..C
007112  004737  017340                  JSR     PC,@#$CC17      ; CHECK FOR CC = 17
007116                                  MFPS    R1              ; MOVE PSW TO R1
007116  106701                          .WORD   106700!..C
007120  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11 [C BIT SHOULD NOT BE EFFECTED BY MFP
007124  022701  177757                  CMP     #177757,R1      ; CHECK TO SEE IF BIT 8 OF PSW WAS EXTENDED THRU R1
007130  001404                          BEQ     MODE0
007132                          EPSW:
007132  012745  000204                  MOV     #204,-(R5)
007136  005245                          INC     -(R5)
007140  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
                                        ;;*******************************************************************************

007142                                  MODE0:
007142  021527  000054                          CMP     (R5),#54
007146  001063                                  BNE     EMODE0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
007150  005215                                  INC     (R5)
007152  112700  000252                          MOVB    #252,R0         ; LOAD REGISTERS
007156  110001                                  MOVB    R0,R1           ;
007160  110102                                  MOVB    R1,R2           ;
007162  122702  000252                          CMPB    #252,R2         ; CHECK IT
007166  001404                                  BEQ     1$              ; OK, CONTINUE
007170  012745  000205                          MOV     #205,-(R5)
007174  005245                                  INC     -(R5)
007176  000000                                  HALT                    ; MOV INSTRUCTION FAILED IN MODE 0
007200  012700  125252                  1$:     MOV     #125252,R0      ; LOAD REGISTERS
007204  010001                                  MOV     R0,R1           ;
007206  010102                                  MOV     R1,R2           ;
007210  022702  125252                          CMP     #125252,R2      ; CHECK IT
007214  001404                                  BEQ     MODE1           ; OK, CONTINUE
007216  012745  000206                          MOV     #206,-(R5)
007222  005245                                  INC     -(R5)
007224  000000                                  HALT                    ; MOV INSTRUCTION FAILED IN MODE 0

007226  012700  000136                  MODE1:  MOV     #TEMP,R0        ; LOAD ADDRESSES INTO REGS.
007232  012701  000140                          MOV     #TEMP1,R1       ;
007236  012702  000142                          MOV     #TEMP2,R2       ;
007242  005067  170674                          CLR     TEMP2           ; START CLEAN
007246  112710  000125                          MOVB    #125,(R0)       ; LOAD THE LOCATIONS
007252  111011                                  MOVB    (R0),(R1)       ; TEMP ---> TEMP1
007254  111112                                  MOVB    (R1),(R2)       ; TEMP1 ---> TEMP2
007256  122767  000125  170656                  CMPB    #125,TEMP2      ; CHECK IT
007264  001404                                  BEQ     1$              ; OK, CONTINUE
007266  012745  000207                          MOV     #207,-(R5)
007272  005245                                  INC     -(R5)
007274  000000                                  HALT                    ; MOV INSTRUCTION FAILED IN MODE 1
007276  012710  052525                  1$:     MOV     #52525,(R0)     ; LOAD THE LOCATIONS
007302  011011                                  MOV     (R0),(R1)       ; TEMP ---> TEMP1
007304  011112                                  MOV     (R1),(R2)       ; TEMP1 ---> TEMP2
007306  022767  052525  170626                  CMP     #52525,TEMP2    ; CHECK IT
007314  001404                                  BEQ     MODE2           ; OK, CONTINUE
007316                                  EMODE0:
007316  012745  000210                          MOV     #210,-(R5)
007322  005245                                  INC     -(R5)
007324  000000                                  HALT                    ; MOV INSTRUCTION FAILED IN MODE 1
                                                                        ; OR WRONG SEQUENCE



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

```
                                ;*TEST: 55      CHECK MODE 2 USING THE MOVB AND MOV INSTRUCTIONS
                                ;;*****************************************************************************

007326                          MODE2:
007326  021527  000055                  CMP     (R5),#55
007332  001050                          BNE     EMODE2          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
007334  005215                          INC     (R5)
007336  012700  000136                  MOV     #TEMP,R0        ; LOAD ADDRESSES
007342  012701  000140                  MOV     #TEMP1,R1       ;
007346  012702  000142                  MOV     #TEMP2,R2
007352  105022                          CLRB    (R2)+           ; START CLEAN
007354  112710  000252                  MOVB    #252,(R0)       ; LOAD THE LOCATIONS
007360  112021                          MOVB    (R0)+,(R1)+     ; TEMP ---> TEMP1
007362  105201                          INCB    R1              ; MAKE IT EVEN
007364  111167  170546                  MOVB    (R1),TEMP       ; MORE 0'S INTO TEMP
007370  105200                          INCB    R0              ; MAKE IT EVEN
007372  112021                          MOVB    (R0)+,(R1)+     ; TEMP1 ---> TEMP2
007374  124227  000252                  CMPB    -(R2),#252      ; CHECK IT
007400  001003                          BNE     1$              ; FAILED
007402  105767  170530                  TSTB    TEMP            ; CHECK IT
007406  001404                          BEQ     2$              ; OK, CONTINUE
007410                          1$:
007410  012745  000211                  MOV     #211,-(R5)
007414  005245                          INC     -(R5)
007416  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 2

007420  005741                  2$:     TST     -(R1)
007422  005022                          CLR     (R2)+           ; START CLEAN
007424  012740  125252                  MOV     #125252,-(R0)   ; LOAD LOCATIONS
007430  012020                          MOV     (R0)+,(R0)+     ; TEMP ---> TEMP1
007432  011067  170500                  MOV     (R0),TEMP       ; 0 ---> TEMP
007436  012121                          MOV     (R1)+,(R1)+     ; 125252 ---> TEMP2
007440  024227  125252                  CMP     -(R2),#125252   ; CHECK IT
007444  001003                          BNE     EMODE2          ; FAILED
007446  005767  170464                  TST     TEMP            ; CHECK IT
007452  001404                          BEQ     MODE3           ; OK, CONTINUE
007454                          EMODE2:
007454  012745  000212                  MOV     #212,-(R5)
007460  005245                          INC     -(R5)
007462  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 2
                                                                ; OR WRONG SEQUENCE
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 56      CHECK MODE 3 USING THE MOVB AND MOV INSTRUCTIONS
                                        ;;*******************************************************************************

007464                                  MODE3:
007464  021527  000056                          CMP     (R5),#56
007470  001066                                  BNE     EMODE3          ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
007472  005215                                  INC     (R5)
007474  012767  000136  170424                  MOV     #TEMP,ADR       ; LOAD ADDRESSES
007502  012767  000140  170420                  MOV     #TEMP1,ADR1     ;
007510  012767  000142  170414                  MOV     #TEMP2,ADR2     ;
007516  012700  000126                          MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
007522  012701  000130                          MOV     #ADR1,R1        ;
007526  105067  170410                          CLRB    TEMP2           ; START CLEAN
007532  112767  000125  170376                  MOVB    #125,TEMP       ;
007540  113031                                  MOVB    @(R0)+,@(R1)+   ; TEMP ---> TEMP1
007542  113167  170370                          MOVB    @(R1)+,TEMP     ; TEMP2 ---> TEMP
007546  113030                                  MOVB    @(R0)+,@(R0)+   ; TEMP1 ---> TEMP2
007550  122767  000125  170364                  CMPB    #125,TEMP2      ; CHECK IT
007556  001003                                  BNE     1$              ; FAILED
007560  105767  170352                          TSTB    TEMP            ; CHECK IT
007564  001404                                  BEQ     2$              ; OK, CONTINUE
007566                                  1$:
007566  012745  000213                          MOV     #213,-(R5)
007572  005245                                  INC     -(R5)
007574  000000                                  HALT                    ; INSTRUCTIONS FAILED IN MODE 3
007576  005067  170340                  2$:     CLR     TEMP2           ; START CLEAN
007602  012767  052525  170326                  MOV     #52525,TEMP     ; LOAD LOCATIONS
007610  012700  000126                          MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
007614  012701  000130                          MOV     #ADR1,R1        ;
007620  013030                                  MOV     @(R0)+,@(R0)+   ; TEMP ---> TEMP1
007622  013067  170310                          MOV     @(R0)+,TEMP     ; TEMP2 ---> TEMP
007626  013131                                  MOV     @(R1)+,@(R1)+   ; TEMP1 ---> TEMP2
007630  022767  052525  170304                  CMP     #52525,TEMP2    ; CHECK IT
007636  001003                                  BNE     EMODE3          ; FAILED
007640  005767  170272                          TST     TEMP            ; CHECK IT
007644  001404                                  BEQ     MODE4           ; OK, CONTINUE
007646                                  EMODE3:
007646  012745  000214                          MOV     #214,-(R5)
007652  005245                                  INC     -(R5)
007654  000000                                  HALT                    ; INSTRUCTIONS FAILED IN MODE 3



                                        ;;*******************************************************************************
                                        ;*TEST: 57      CHECK MODE 4 USING THE MOVB AND MOV INSTRUCTIONS
                                        ;;*******************************************************************************

007656                                  MODE4:
007656  021527  000057                          CMP     (R5),#57
007662  001120                                  BNE     EMODE4          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
007664  005215                                  INC     (R5)
007666  105067  170244                          CLRB    TEMP            ; START CLEAN
007672  012700  000136                          MOV     #TEMP,R0        ; LOAD ADDRESSES
007676  012701  000140                          MOV     #TEMP1,R1       ;
007702  012702  000142                          MOV     #TEMP2,R2       ;
```

```
007706  005202                          INC     R2                      ; ADJUST THE POINTER
007710  021267  170227                  CMP     (R2),TEMP2+1
007714  001404                          BEQ     1$
007716  012745  000215                  MOV     #215,-(R5)
007722  005245                          INC     -(R5)
007724  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 4
007726  112742  000252          1$:     MOVB    #252,-(R2)              ; LOAD TEMP2
007732  005201                          INC     R1                      ; ADJUST THE POINTERS
007734  005202                          INC     R2                      ;
007736  114241                          MOVB    -(R2),-(R1)             ; TEMP2 ---> TEMP1
007740  005200                          INC     R0                      ; ADJUST THE POINTERS
007742  005202                          INC     R2                      ;
007744  114042                          MOVB    -(R0),-(R2)             ; TEMP ---> TEMP2
007746  105200                          INCB    R0                      ; ADJUST THE POINTERS
007750  021067  170163                  CMP     (R0),TEMP+1
007754  001404                          BEQ     2$
007756  012745  000216                  MOV     #216,-(R5)
007762  005245                          INC     -(R5)
007764  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 4
007766  105201                  2$:     INCB    R1                      ;
007770  114140                          MOVB    -(R1),-(R0)             ; TEMP1 ---> TEMP
007772  122767  000252  170136          CMPB    #252,TEMP               ; CHECK IT
010000  001003                          BNE     3$                      ; FAILED
010002  105767  170134                  TSTB    TEMP2                   ; CHECK IT
010006  001404                          BEQ     4$                      ; OK, CONTINUE
010010                          3$:
010010  012745  000217                  MOV     #217,-(R5)
010014  005245                          INC     -(R5)
010016  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 4
010020  005067  170112          4$:     CLR     TEMP                    ; START CLEAN
010024  012700  000136                  MOV     #TEMP,R0                ; LOAD ADDRESSES
010030  012701  000140                  MOV     #TEMP1,R1               ;
010034  012702  000142                  MOV     #TEMP2,R2               ;
010040  005722                          TST     (R2)+                   ; ADJUST THE POINTER
010042  021267  170076                  CMP     (R2),TEMP2+2
010046  001404                          BEQ     5$
010050  012745  000220                  MOV     #220,-(R5)
010054  005245                          INC     -(R5)
010056  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 4
010060  012742  125252          5$:     MOV     #125252,-(R2)           ; LOAD TEMP2
010064  005721                          TST     (R1)+                   ; ADJUST THE POINTERS
010066  005722                          TST     (R2)+                   ;
010070  014241                          MOV     -(R2),-(R1)             ; TEMP2 ---> TEMP1
010072  005720                          TST     (R0)+                   ; ADJUST POINTERS
010074  005722                          TST     (R2)+                   ;
010076  014042                          MOV     -(R0),-(R2)             ; TEMP ---> TEMP2
010100  005720                          TST     (R0)+                   ; ADJUST THE POINTERS
010102  005721                          TST     (R1)+                   ;
010104  014140                          MOV     -(R1),-(R0)             ; TEMP1 ---> TEMP
010106  022767  125252  170022          CMP     #125252,TEMP            ; CHECK IT
010114  001003                          BNE     EMODE4                  ; FAILED
010116  005767  170020                  TST     TEMP2                   ; CHECK IT
010122  001404                          BEQ     MODE5                   ; OK, CONTINUE
010124                          EMODE4:
010124  012745  000221                  MOV     #221,-(R5)
010130  005245                          INC     -(R5)
```

```
2208  010132  000000                    HALT                    ; INSTRUCTIONS FAILED IN MODE 4
2209                                                            ; OR WRONG SEQUENCE
```

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

010134                          MODE5:
010134  021527  000060                  CMP     (R5),#60
010140  001105                          BNE     EMODE5          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
010142  005215                          INC     (R5)
010144  105067  167766                  CLRB    TEMP            ; START CLEAN
010150  012767  000136  167750          MOV     #TEMP,ADR       ; LOAD ADDRESSES
010156  012767  000140  167744          MOV     #TEMP1,ADR1     ;
010164  012767  000142  167740          MOV     #TEMP2,ADR2     ;
010172  012700  000126                  MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
010176  012701  000130                  MOV     #ADR1,R1        ;
010202  012702  000132                  MOV     #ADR2,R2        ;
010206  005722                          TST     (R2)+           ; ADJUST THE POINTER
010210  112752  000125                  MOVB    #125,@-(R2)     ; LOAD TEMP2
010214  022122                          CMP     (R1)+,(R2)+     ; ADJUST THE POINTERS
010216  115251                          MOVB    @-(R2),@-(R1)   ; TEMP2 ---> TEMP1
010220  022022                          CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
010222  115052                          MOVB    @-(R0),@-(R2)   ; TEMP ---> TEMP2
010224  022022                          CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
010226  125052                          CMPB    @-(R0),@-(R2)   ;CHECK IT
010230  001404                          BEQ     1$
010232  012745  000222                  MOV     #222,-(R5)
010236  005245                          INC     -(R5)
010240  000000                          HALT
010242  022120                  1$:     CMP     (R1)+,(R0)+     ; ADJUST THE POINTERS
010244  115150                          MOVB    @-(R1),@-(R0)   ; TEMP1 ---> TEMP
010246  122767  000125  167662          CMPB    #125,TEMP       ; CHECK IT
010254  001003                          BNE     2$              ; FAILED
010256  105767  167660                  TSTB    TEMP2           ; CHECK IT
010262  001404                          BEQ     3$              ; OK, CONTINUE
010264                          2$:
010264  012745  000223                  MOV     #223,-(R5)
010270  005245                          INC     -(R5)
010272  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 5
010274  005067  167636          3$:     CLR     TEMP            ; START CLEAN
010300  012700  000126                  MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
010304  012701  000130                  MOV     #ADR1,R1        ;
010310  012702  000132                  MOV     #ADR2,R2        ;
010314  005722                          TST     (R2)+           ; ADJUST THE POINTEF
010316  012752  052525                  MOV     #52525,@-(R2)   ; LOAD TEMP2
010322  022122                          CMP     (R1)+,(R2)+     ; ADJUST THE POINTERS
010324  015251                          MOV     @-(R2),@-(R1)   ; TEMP2 ---> TEMP1
010326  022022                          CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
010330  015052                          MOV     @-(R0),@-(R2)   ; TEMP ---> TEMP2
010332  022021                          CMP     (R0)+,(R1)+     ; ADJUST THE POINTERS
010334  015150                          MOV     @-(R1),@-(R0)   ; TEMP1 ---> TEMP
010336  022767  052525  167572          CMP     #52525,TEMP     ; CHECK IT
010344  001003                          BNE     EMODE5          ; FAILED
010346  005767  167570                  TST     TEMP2           ; CHECK IT
010352  001404                          BEQ     MODE6           ; OK, CONTINUE
010354                          EMODE5:
010354  012745  000224                  MOV     #224,-(R5)
010360  005245                          INC     -(R5)
```

```
010362  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 5
                                                                ; OR WRONG SEQUENCE



                                ;;*******************************************************************
                                ;*TEST: 61       CHECK MODE 6 USING THE MOVB AND MOV INSTRUCTIONS
                                ;;*******************************************************************

010364                          MODE6:
010364  021527  000061                  CMP     (R5),#61
010370  001055                          BNE     EMODE6          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
010372  005215                          INC     (R5)
010374  005067  167542                  CLR     TEMP2           ; START CLEAN
010400  012700  000136                  MOV     #TEMP,R0        ; LOAD ADDRESSES
010404  012701  000140                  MOV     #TEMP1,R1       ;
010410  012702  000142                  MOV     #TEMP2,R2       ;
010414  112760  000252  000000          MOVB    #252,0(R0)      ; LOAD TEMP (LOW BYTE)
010422  112760  000252  000001          MOVB    #252,1(R0)      ; LOAD TEMP (HIGH BYTE)
010430  022767  125252  167500          CMP     #125252,TEMP    ; CHECK IT
010436  001012                          BNE     1$              ; FAILED
010440  116062  000001  000000          MOVB    1(R0),0(R2)     ; TEMP(H) ---> TEMP2(L)
010446  116160  000002  000005          MOVB    2(R1),5(R0)     ; TEMP2(L) ---> TEMP2(H)
010454  022767  125252  167460          CMP     #125252,TEMP2   ; CHECK IT
010462  001404                          BEQ     2$              ; OK, CONTINUE
010464                          1$:
010464  012745  000225                  MOV     #225,-(R5)
010470  005245                          INC     -(R5)
010472  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 6
010474  005067  167440          2$:     CLR     TEMP1           ; START CLEAN
010500  012760  052525  000000          MOV     #52525,0(R0)    ; LOAD TEMP
010506  016260  177774  000002          MOV     -4(R2),2(R0)    ; TEMP ---> TEMP1
010514  022767  052525  167416          CMP     #52525,TEMP1    ; CHECK IT
010522  001404                          BEQ     MODE7           ; OK, CONTINUE
010524                          EMODE6:
010524  012745  000226                  MOV     #226,-(R5)
010530  005245                          INC     -(R5)
010532  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 6
                                                                ; OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************
                                ;*TEST: 62      CHECK MODE 7 USING THE MOVB AND MOV INSTRUCTIONS
                                ;;*******************************************************************

010534                          MODE7:
010534  021527  000062                  CMP     (R5),#62
010540  001052                          BNE     EMODE7          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
010542  005215                          INC     (R5)
010544  005067  167370                  CLR     TEMP1           ; START CLEAN
010550  012767  000136  167350          MOV     #TEMP,ADR       ; LOAD ADDRESSES
010556  012767  000140  167344          MOV     #TEMP1,ADR1     ;
010564  012767  000142  167340          MOV     #TEMP2,ADR2     ;
010572  012700  000126                  MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
010576  012701  000130                  MOV     #ADR1,R1        ;
010602  012702  000132                  MOV     #ADR2,R2        ;
010606  112770  000252  000000          MOVB    #252,@0(R0)     ; LOAD TEMP
010614  117270  177774  000002          MOVB    @-4(R2),@2(R0)  ; TEMP ---> TEMP1
010622  122767  000252  167310          CMPB    #252,TEMP1      ; CHECK IT
010630  001404                          BEQ     1$              ; OK, CONTINUE
010632  012745  000227                  MOV     #227,-(R5)
010636  005245                          INC     -(R5)
010640  000000                          HALT                    ; MODE 7 IS FAILING
010642  012770  125252  000000  1$:     MOV     #125252,@0(R0)  ; LOAD TEMP
010650  017270  177774  000002          MOV     @-4(R2),@2(R0)  ; TEMP ---> TEMP1
010656  022767  125252  167254          CMP     #125252,TEMP1   ; CHECK IT
010664  001404                          BEQ     TSTB1           ; OK, CONTINUE
010666                          EMODE7:
010666  012745  000230                  MOV     #230,-(R5)
010672  005245                          INC     -(R5)
010674  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 7
                                                                ; OR WRONG SEQUENCE
```

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


                                        ;;*******************************************************************************
                                        ;*TEST: 63       NEW INSTRUCTIONS USED IN THIS SECTION ARE TSTB, CLRB, MOVB
                                        ;;*******************************************************************************

010676                                  TSTB1:
010676  021527  000063                          CMP     (R5),#63
010702  001042                                  BNE     ETSTB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF TEST
010704  005215                          2$:     INC     (R5)
010706  012700  000136                          MOV     #TEMP,R0        ; LOAD ADDRESSES
010712  012701  000140                          MOV     #TEMP1,R1       ;
010716  000277                                  SCC
010720  105010                                  CLRB    (R0)            ; CLEAR THE LOCATION
010722  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
010726  105710                                  TSTB    (R0)            ; CHECK IT
010730  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
010734  112711  000377                          MOVB    #377,(R1)       ; LOAD THE LOCATION
010740  004737  017236                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
010744  105711                                  TSTB    (R1)            ; CHECK IT
010746  004737  017236                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
010752  010002                                  MOV     R0,R2           ; R2 IS NOW POINTING TO LOCATION TEMP
010754  112762  000200  000000                  MOVB    #200,0(R2)      ; PLACE #200 IN LOCATION TEMP
010762  112241                                  MOVB    (R2)+,-(R1)     ; MOVE #200 TO LOCATION TEMP+1
010764  026127  177777  100200                  CMP     -1(R1),#100200  ; CHECK THE DATA IN LOCATION TEMP
010772  001404                                  BEQ     4$
010774  012745  000231                          MOV     #231,-(R5)
011000  005245                                  INC     -(R5)
011002  000000                                  HALT                    ; MOVB INSTRUCTION FAILED
011004  020102                          4$:     CMP     R1,R2           ; CHECK THE REGISTERS FOR PROPER VALUE
011006  001404                                  BEQ     CMPB1
011010                                  ETSTB1:
011010  012745  000232                          MOV     #232,-(R5)
011014  005245                                  INC     -(R5)
011016  000000                                  HALT                    ; MOVB INSTRUCTION FAILED OR WRONG SEQUENCE
```

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

011020                          CMPB1:
011020  021527  000064                  CMP     (R5),#64
011024  001032                          BNE     ECMPB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
011026  005215                  1$:     INC     (R5)
011030  012701  000142                  MOV     #TEMP2,R1
011034  012702  000136                  MOV     #TEMP,R2        ; LOAD ADDRESS
011040  012711  000077                  MOV     #77,(R1)        ; PLACE 77 IN LOCATION TEMP2
011044  112704  000377                  MOVB    #377,R4         ; R4 SHOULD CONTAIN #177777
011050  150412                          BISB    R4,(R2)         ; LOAD LOCATION
011052  004737  017236                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
011056  120412                          CMPB    R4,(R2)         ; CHECK COMPARE
011060  001404                          BEQ     2$              ; CONTINUE IF OK
011062  012745  000233                  MOV     #233,-(R5)
011066  005245                          INC     -(R5)
011070  000000                          HALT                    ; BISB OR CMPB INSTRUCTION FAILED
011072  121112                  2$:     CMPB    (R1),(R2)       ; CHECK IT AGAIN
011074  100004                          BPL     3$              ; CONTINUE IF OK
011076  012745  000234                  MOV     #234,-(R5)
011102  005245                          INC     -(R5)
011104  000000                          HALT                    ; CMPB INSTRUCTION FAILED [WRONG CC]
011106  121211                  3$:     CMPB    (R2),(R1)       ; ONCE MORE
011110  100404                          BMI     BICB1           ; CONTINUE IF OK
011112                          ECMPB1:
011112  012745  000235                  MOV     #235,-(R5)
011116  005245                          INC     -(R5)
011120  000000                          HALT


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

011122                          BICB1:
011122  021527  000065                  CMP     (R5),#65
011126  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
011130  012745  000236                  MOV     #236,-(R5)
011134  005245                          INC     -(R5)
011136  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
011140  005215                  2$:     INC     (R5)
011142  012703  000136                  MOV     #TEMP,R3        ; LOAD ADDRESS
011146  112713  000377                  MOVB    #377,(R3)       ; LOAD LOCATION
011152  012700  000140                  MOV     #TEMP1,R0       ;PLACE THE ADDRESS OF LOCATION TEMP1 IN R0
011156  010001                          MOV     R0,R1           ;AND R1
011160  112721  000252                  MOVB    #252,(R1)+      ; PLACE #252 IN TEMP1
011164  000277                          SCC
011166  146013  000000                  BICB    0(R0),(R3)      ; CLEAR EVERY OTHER BIT
011172  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
011176  136113  177777                  BITB    -1(R1),(R3)     ; CHECK IT
011202  001404                          BEQ     4$              ; CONTINUE IF OK
011204  012745  000237                  MOV     #237,-(R5)
011210  005245                          INC     -(R5)
```

```
011212  000000                          HALT                    ; BICB OR BITB INSTRUCTION FAILED
011214  132713  000125          4$:     BITB    #125,(R3)       ; CHECK IT
011220  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
011224  154113                          BISB    -(R1),(R3)      ; SET THE BITS THAT WERE CLEARED
011226  100404                          BMI     6$              ; CONTINUE IF OK
011230  012745  000240                  MOV     #240,-(R5)
011234  005245                          INC     -(R5)
011236  000000                          HALT                    ; BITB OR BISB INSTRUCTION FAILED
011240  012746  000177          6$:     MOV     #177,-(SP)      ; STORE #177 ON THE STACK
011244  142613                          BICB    (SP)+,(R3)      ; CLEAR ALL THE BITS EXCEPT SIGN BIT
011246  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
011252  132713  000377                  BITB    #377,(R3)       ; CHECK IT
011256  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
011262  010300                          MOV     R3,R0           ; PLACE THE ADDRESS OF LOCATION TEMP IN R0
011264  012710  000140                  MOV     #TEMP1,(R0)     ; PLACE THE ADDRESS OF LOCATION TEMP1 IN TEMP
011270  012730  000377                  MOV     #377,@(R0)+     ; WRITE A 377 IN LOCATION TEMP1
011274  000263                          SEVC                    ; SET V & C BITS
011276  145070  000000                  BICB    @-(R0),@(R0)    ; BIT CLEAR THE CONTENTS
                                                                ; OF TEMP1 TO THE CONTENTS OF TEMP1
011302  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
011306  022027  000140                  CMP     (R0)+,#TEMP1    ; MAKE SURE THAT (R0) IS POINTING TO LOCATION TEMP1
011312  001404                          BEQ     8$
011314  012745  000241                  MOV     #241,-(R5)
011320  005245                          INC     -(R5)
011322  000000                          HALT                    ; BICB OR CMP INSTRUCTION FAILED IN THE SPECIFIC MODE
011324  005750                  8$:     TST     @-(R0)          ; TEST LOCATION TEMP1
011326  001404                          BEQ     10$
011330  012745  000242                  MOV     #242,-(R5)
011334  005245                          INC     -(R5)
011336  000000                          HALT                    ; BICB INSTRUCTION FAILED
011340  000257                  10$:    CCC
011342  141010                          BICB    (R0),(R0)       ; CLEAR THE LOCATION TEMP
011344  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 66      NEW INSTRUCTIONS USED IN THIS SECTION ARE INCB, DECB
                                        ;;*******************************************************************************

011350                                  INCB1:
011350  021527  000066                          CMP     (R5),#66
011354  001067                                  BNE     EINCB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
011356  005215                          1$:     INC     (R5)
011360  012704  000136                          MOV     #TEMP,R4        ; LOAD ADDRESS
011364  112714  000177                          MOVB    #177,(R4)       ; TEMP LOCATION=177
011370  000261                                  SEC
011372  105214                                  INCB    (R4)            ; ADD ONES INTO LOCATION
011374  004737  017320                          JSR     PC,@#$CC13      ; CHECK FOR CC = 13
011400  012714  000376                          MOV     #376,(R4)
011404  012700  017256                          MOV     #$CC11,R0       ; MAKE R0 POINT TO CHECKING ROUTINE FOR CC = 11
011410  105224                                  INCB    (R4)+           ;
011412  004720                                  JSR     PC,(R0)+        ; CHECK FOR CC = 11
011414  105744                                  TSTB    -(R4)           ; DECREMENT R4 BY 1
011416  005746                                  TST     -(SP)           ; AND SP BY 2
011420  010426                                  MOV     R4,(SP)+        ; PLACE THE ADDRESS OF TEMP ON THE STACK
011422  000241                                  CLC                     ; CLEAR C BIT
011424  105256                                  INCB    @-(SP)          ; INCREMENT THE CONTENTS OF LOCATION TEMP
011426  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
011432  123634                                  CMPB    @(SP)+,@(R4)+   ; RESTORE STACK POINTER
011434  000261                                  SEC                     ; SET C BIT
011436  105264  177777                          INCB    -1(R4)
011442  004737  017066                          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
011446  124427  000001                          CMPB    -(R4),#1        ; CHECK IT
011452  001404                                  BEQ     2$              ; CONTINUE IF OK
011454  012745  000243                          MOV     #243,-(R5)
011460  005245                                  INC     -(R5)
011462  000000                                  HALT                    ; INCB INSTRUCTION FAILED
011464  000261                          2$:     SEC
011466  105314                                  DECB    (R4)            ; SUBTRACT ONES FROM LOCATION
011470  004737  017172                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
011474  105324                                  DECB    (R4)+
011476  004740                                  JSR     PC,-(R0)        ; CHECK FOR CC = 11
011500  112764  000200  177777                  MOVB    #200,-1(R4)
011506  105344                                  DECB    -(R4)           ;
011510  004760  177650                          JSR     PC,$CC3-$CC11(R0); CHECK FOR CC = 3
011514  105364  000000                          DECB    0(R4)           ;
011520  004737  017066                          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
011524  126427  000000  000176                  CMPB    0(R4),#176
011532  001404                                  BEQ     COMB1
011534                                  EINCB1:
011534  012745  000244                          MOV     #244,-(R5)
011540  005245                                  INC     -(R5)
011542  000000                                  HALT                    ; DECB INSTRUCTION FAILED OR SEQUENCE ERROR
```

```
                                        ;;*******************************************************************
                                        ;*TEST: 67      NEW INSTRUCTION IN THIS SECTION IS COMB
                                        ;;*******************************************************************

011544                                  COMB1:
011544  021527  000067                          CMP     (R5),#67
011550  001404                                  BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
011552  012745  000245                          MOV     #245,-(R5)
011556  005245                                  INC     -(R5)
011560  000000                                  HALT                    ; TEST IS IN WRONG SEQUENCE
011562  005215                          1$:     INC     (R5)
011564  012703  000136                          MOV     #TEMP,R3        ; LOAD ADDRESS
011570  012704  000140                          MOV     #TEMP1,R4
011574  012714  000252                          MOV     #252,(R4)
011600  112413                                  MOVB    (R4)+,(R3)      ; LOAD EVERY OTHER BIT
011602  000277                                  SCC
011604  105113                                  COMB    (R3)            ; 1'S COMPLEMENT
011606  004737  017066                          JSR     PC,@#$CC1       ; CHECK FOR CC = 1
011612  122713  000125                          CMPB    #125,(R3)       ; CHECK IT
011616  001404                                  BEQ     2$              ; CONTINUE IF OK
011620  012745  000246                          MOV     #246,-(R5)
011624  005245                                  INC     -(R5)
011626  000000                                  HALT                    ; COMB INSTRUCTION FAILED
011630  000277                          2$:     SCC
011632  105113                                  COMB    (R3)            ; COMPLEMENT BACK
011634  004737  017256                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
011640  010400                                  MOV     R4,R0
011642  126013  177777                          CMPB    -1(R0),(R3)     ; CHECK IT
011646  001404                                  BEQ     3$              ; CONTINUE IF OK
011650  012745  000247                          MOV     #247,-(R5)
011654  005245                                  INC     -(R5)
011656  000000                                  HALT                    ; COMB INSTRUCTION FAILED
011660  112724  000377                  3$:     MOVB    #377,(R4)+
011664  114413                                  MOVB    -(R4),(R3)      ; PLACE #377 IN (R3)
011666  000277                                  SCC
011670  105113                                  COMB    (R3)
011672  004737  017172                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

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

                                NEGB1:
011676                                  CMP     (R5),#70
011676  021527  000070                  BNE     ENEGB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
011702  001027                  1$:     INC     (R5)
011704  005215                          MOV     #TEMP,R0        ; LOAD ADDRESS
011706  012700  000136                  MOVB    #1,(R0)         ; LOAD THE LOCATION
011712  112710  000001                  NEGB    (R0)            ; 2'S COMPLEMENT
011716  105410                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
011720  004737  017256                  CMPB    #377,(R0)       ; CHECK IT
011724  122710  000377                  BEQ     2$              ; CONTINUE IF OK
011730  001404                          MOV     #250,-(R5)
011732  012745  000250                  INC     -(R5)
011736  005245                          HALT                    ; NEGB INSTRUCTION FAILED
011740  000000                  2$:     MOV     #200,(R0)
011742  012710  000200                  NEGB    (R0)            ; 2'S COMPLEMENT
011746  105410                          JSR     PC,@#$CC13      ; CHECK FOR CC = 13
011750  004737  017320                  CMPB    #200,(R0)       ; CHECK IT
011754  122710  000200                  BEQ     ROLB1           ; CONTINUE IF OK
011760  001404                  ENEGB1:
011762                                  MOV     #251,-(R5)
011762  012745  000251                  INC     -(R5)
011766  005245                          HALT                    ; WRONG RESULT AT TEMP OR WRONG SEQUENCE
011770  000000


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

                                ROLB1:
011772                                  CMP     (R5),#71
011772  021527  000071                  BNE     EROLB1          ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
011776  001030                          INC     (R5)
012000  005215                          MOV     #TEMP1,R1       ; LOAD ADDRESS
012002  012701  000140                  MOVB    #40,(R1)        ; LOAD LOCATION
012006  112711  000040                  CCC                     ; CLEAR FLAGS
012012  000257                          ROLB    (R1)            ; SHIFT
012014  106111                          ROLB    (R1)            ;
012016  106111                          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
012020  004737  017300                  CMPB    #200,(R1)       ; CHECK IT
012024  122711  000200                  BEQ     1$              ; CONTINUE IF OK
012030  001404                          MOV     #252,-(R5)
012032  012745  000252                  INC     -(R5)
012036  005245                          HALT                    ; ROLB INSTRUCTION FAILED
012040  000000                  1$:     ROLB    (R1)            ; SHIFT
012042  106111                          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
012044  004737  017214                  ROLB    (R1)            ; SHIFT
012050  106111                          CMPB    #1,(R1)         ; CHECK IT
012052  122711  000001                  BEQ     RORB1           ; CONTINUE IF OK
012056  001404                  EROLB1:
012060                                  MOV     #253,-(R5)
012060  012745  000253                  INC     -(R5)
012064  005245
```

```
2616  012066  000000                    HALT                    ; WRONG RESULT AT TEMP1 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;*TEST: 72      NEW INSTRUCTION IN THIS SECTION IS RORB
                                ;;*******************************************************************************

                                RORB1:
012070                                  CMP     (R5),#72
012070  021527  000072                  BNE     ERORB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
012074  001030                          INC     (R5)
012076  005215                          MOV     #TEMP1,R2       ; LOAD ADDRESS
012100  012702  000140                  MOVB    #4,(R2)         ; LOAD LOCATION
012104  112712  000004                  CCC                     ; CLEAR FLAGS
012110  000257                          RORB    (R2)            ; SHIFT
012112  106012                          RORB    (R2)            ;
012114  106012                          CMPB    #1,(R2)         ; CHECK IT
012116  122712  000001                  BEQ     1$              ; CONTINUE IF OK
012122  001404                          MOV     #254,-(R5)
012124  012745  000254                  INC     -(R5)
012130  005245                          HALT                    ; RORB INSTRUCTION FAILED
012132  000000                  1$:     RORB    (R2)            ; SHIFT
012134  106012                          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
012136  004737  017214                  RORB    (R2)            ; SHIFT
012142  106012                          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
012144  004737  017300                  CMPB    #200,(R2)       ; CHECK IT
012150  122712  000200                  BEQ     ASLB1           ; CONTINUE IF OK
012154  001404                  ERORB1:
012156                                  MOV     #255,-(R5)
012156  012745  000255                  INC     -(R5)
012162  005245                          HALT                    ; WRONG RESULT AT TEMP1 OR WRONG SEQUENCE
012164  000000


                                ;;*******************************************************************************
                                ;*TEST: 73      NEW INSTRUCTION IN THIS SECTION IS ASLB
                                ;;*******************************************************************************

                                ASLB1:
012166                                  CMP     (R5),#73
012166  021527  000073                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
012172  001404                          MOV     #256,-(R5)
012174  012745  000256                  INC     -(R5)
012200  005245                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
012202  000000                  2$:     INC     (R5)
012204  005215                          MOV     #TEMP1,R3       ; LOAD ADDRESS
012206  012703  000140                  MOVB    #40,(R3)        ; LOAD LOCATION
012212  112713  000040                  CCC                     ; CLEAR FLAGS
012216  000257                          ASLB    (R3)            ; SHIFT
012220  106313                          ASLB    (R3)            ;
012222  106313                          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
012224  004737  017300                  CMPB    #200,(R3)       ; CHECK IT
012230  122713  000200                  BEQ     4$              ; CONTINUE IF OK
012234  001404                          MOV     #257,-(R5)
012236  012745  000257                  INC     -(R5)
012242  005245                          HALT                    ; ASLB INSTRUCTION FAILED
012244  000000                  4$:     ASLB    (R3)            ; SHIFT
012246  106313                          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
012250  004737  017214                  ASLB    (R3)            ; SHIFT
012254  106313
```

```
2673  012256  004737  017150          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
```

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

012262                          ASRB1:
012262  021527  000074                  CMP     (R5),#74
012266  001040                          BNE     EASRB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
012270  005215                  1$:     INC     (R5)
012272  012704  000140                  MOV     #TEMP1,R4       ; LOAD ADDRESSES
012276  012703  000142                  MOV     #TEMP2,R3       ;
012302  112714  000004                  MOVB    #4,(R4)         ; LOAD LOCATION
012306  000257                          CCC                     ; CLEAR FLAGS
012310  106214                          ASRB    (R4)            ; SHIFT
012312  106214                          ASRB    (R4)            ;
012314  122714  000001                  CMPB    #1,(R4)         ; CHECK IT
012320  001404                          BEQ     2$              ; CONTINUE IF OK
012322  012745  000260                  MOV     #260,-(R5)
012326  005245                          INC     -(R5)
012330  000000                          HALT                    ; ASRB INSTRUCTION FAILED
012332  106214                  2$:     ASRB    (R4)            ; SHIFT
012334  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
012340  106214                          ASRB    (R4)            ; SHIFT
012342  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
012346  112713  000202                  MOVB    #202,(R3)       ; LOAD LOCATION
012352  106213                          ASRB    (R3)            ; SHIFT
012354  106213                          ASRB    (R3)            ;
012356  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
012362  122713  000340                  CMPB    #340,(R3)       ; CHECK IT
012366  001404                          BEQ     ADCB1           ; CONTINUE IF OK
012370                          EASRB1:
012370  012745  000261                  MOV     #261,-(R5)
012374  005245                          INC     -(R5)
012376  000000                          HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE


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

012400                          ADCB1:
012400  021527  000075                  CMP     (R5),#75
012404  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
012406  012745  000262                  MOV     #262,-(R5)
012412  005245                          INC     -(R5)
012414  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
012416  005215                  2$:     INC     (R5)
012420  012700  000142                  MOV     #TEMP2,R0       ; LOAD ADDRESS
012424  105010                          CLRB    (R0)            ; CLEAR THE LOCATION
012426  000257                          CCC                     ; CLEAR FLAGS
012430  105510                          ADCB    (R0)            ; ADD C BIT = 0
012432  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
012436  000261                          SEC                     ; C=1
012440  105510                          ADCB    (R0)            ; ADD C BIT=1
012442  000261                          SEC                     ; C=1
012444  105510                          ADCB    (R0)            ; AGAIN
```

```
012446  004737  017046                JSR     PC,@#$CC0       ; CHECK FOR CC = 0
012452  122710  000002                CMPB    #2,(R0)         ; CHECK IT
012456  001404                        BEQ     4$              ; CONTINUE IF OK
012460  012745  000263                MOV     #263,-(R5)
012464  005245                        INC     -(R5)
012466  000000                        HALT                    ; ADCB INSTRUCTION FAILED
012470  112710  000177          4$:   MOVB    #177,(R0)       ; LOAD LARGEST POSITVE BYTE
012474  000261                        SEC                     ; C=1
012476  105510                        ADCB    (R0)            ; ADD C BIT=1
012500  004737  017300                JSR     PC,@#$CC12      ; CHECK FOR CC = 12
012504  122710  000200                CMPB    #200,(R0)       ; CHECK IT
012510  001404                        BEQ     6$              ; CONTINUE IF OK
012512  012745  000264                MOV     #264,-(R5)
012516  005245                        INC     -(R5)
012520  000000                        HALT                    ; ADCB INSTRUCTION FAILED
012522  112710  000377          6$:   MOVB    #377,(R0)       ; LOAD -1
012526  000261                        SEC                     ; C=1
012530  105510                        ADCB    (R0)            ; ADD C BIT=1
012532  004737  017172                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

```
                                ;;*******************************************************************************
                                ;*TEST: 76      NEW INSTRUCTION IN THIS SECTION IS SBCB
                                ;;*******************************************************************************

012536                          SBCB1:
012536  021527  000076                  CMP     (R5),#76
012542  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
012544  012745  000265                  MOV     #265,-(R5)
012550  005245                          INC     -(R5)
012552  000000                          HALT                    ; TEST IS IN WRONG SEQUENCE
012554  005215                  1$:     INC     (R5)
012556  012701  000142                  MOV     #TEMP2,R1       ; LOAD ADDRESS
012562  112711  000003                  MOVB    #3,(R1)         ; LOAD LOCATION
012566  000257                          CCC                     ; CLEAR FLAGS
012570  105611                          SBCB    (R1)            ; SUBTRACT C BIT=0
012572  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
012576  122711  000003                  CMPB    #3,(R1)         ; CHECK IT
012602  001404                          BEQ     2$              ; CONTINUE IF OK
012604  012745  000266                  MOV     #266,-(R5)
012610  005245                          INC     -(R5)
012612  000000                          HALT                    ; SBCB INSTRUCTION FAILED
012614  000261                  2$:     SEC                     ; C=1
012616  105611                          SBCB    (R1)            ; SUBTRACT C BIT=1
012620  000261                          SEC                     ; C=1
012622  105611                          SBCB    (R1)            ;
012624  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
012630  122711  000001                  CMPB    #1,(R1)         ; CHECK IT
012634  001404                          BEQ     3$              ; CONTINUE IF OK
012636  012745  000267                  MOV     #267,-(R5)
012642  005245                          INC     -(R5)
012644  000000                          HALT                    ; SBCB INSTRUCTION FAILED
012646  000261                  3$:     SEC                     ; C=1
012650  105611                          SBCB    (R1)            ; SUBTRACT C BIT=1
012652  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
012656  000261                          SEC                     ; C=1
012660  105611                          SBCB    (R1)            ; SUBTRACT C BIT = 1
012662  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
012666  122711  000377                  CMPB    #377,(R1)       ; CHECK IT
012672  001404                          BEQ     4$              ; CONTINUE IF OK
012674  012745  000270                  MOV     #270,-(R5)
012700  005245                          INC     -(R5)
012702  000000                          HALT                    ; SBCB INSTRUCTION FAILED
012704  112711  000200          4$:     MOVB    #200,(R1)       ; LOAD R1
```

```
012710  000261                  SEC                     ; C=1
012712  105611                  SBCB    (R1)            ; SUBTRACT C BIT = 1
012714  004737  017106          JSR     PC,@#$CC2       ; CHECK FOR CC = 2
```

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


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

                                TST1:
012720                                  CMP     (R5),#77
012720  021527  000077                  BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
012724  001404                          MOV     #271,-(R5)
012726  012745  000271                  INC     -(R5)
012732  005245                          HALT                    ; TEST IS IN A WRONG SEQUENCE
012734  000000                  1$:     INC     (R5)
012736  005215                          MOV     #TEMP,R1        ; LOAD ADDRESSES
012740  012701  000136                  MOV     #TEMP1,R0       ;
012744  012700  000140                  SCC
012750  000277                          CLR     (R0)            ; CLEAR THE LOCATION
012752  005010                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
012754  004737  017150                  TST     (R0)+           ; CHECK IT
012760  005720                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
012762  004737  017150                  MOV     R0,-(R0)
012766  010040                          MOV     #177777,@(R0)+
012770  012730  177777                  MOV     @-2(R0),(R1)    ; LOAD THE LOCATION
012774  017011  177776                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
013000  004737  017236                  TST     (R1)            ; CHECK IT
013004  005711                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
013006  004737  017236
```

```
                                ;;*******************************************************************
                                ;*TEST: 100     NEW INSTRUCTIONS USED IN THIS SECTION ARE CMP, BIS
                                ;;*******************************************************************

013012                          CMP1:
013012  021527  000100                  CMP     (R5),#100
013016  001113                          BNE     ECMP1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE EST
013020  005215                  1$:     INC     (R5)
013022  012702  000140                  MOV     #TEMP1,R2       ; LOAD ADDRESS
013026  012700  000136                  MOV     #TEMP,R0        ; PLACE THE ADDRESS OF TEMP IN R0
013032  012720  177777                  MOV     #177777,(R0)+   ; PLACE #177777 IN LOCATION TEMP AND INC. R0 BY 2
013036  054012                          BIS     -(R0),(R2)      ; LOAD LOCATION
013040  004737  017236                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
013044  022227  177777                  CMP     (R2)+,#177777   ; CHECK COMPARE
013050  001404                          BEQ     2$              ; CONTINUE IF OK
013052  012745  000272                  MOV     #272,-(R5)
013056  005245                          INC     -(R5)
013060  000000                          HALT                    ; CMP OR BIS INSTRUCTION FAILED
013062  020227  000142          2$:     CMP     R2,#TEMP1+2     ; CHECK R2 TO CONTAIN ADDRESS OF TEMP1+2
013066  001404                          BEQ     3$
013070  012745  000273                  MOV     #273,-(R5)
013074  005245                          INC     -(R5)
013076  000000                          HALT                    ; NO AUTO INCREMENT
013100  022742  000077          3$:     CMP     #77,-(R2)       ; CHECK IT AGAIN
013104  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
013110  022722  077777                  CMP     #77777,(R2)+
013114  004737  017320                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
013120  024227  077777                  CMP     -(R2),#77777    ; ONCE MORE
013124  004737  017236                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
013130  012767  052525  165004          MOV     #52525,TEMP2    ; SET EVERY OTHER BIT IN TEMP2
013136  012767  000142  164774          MOV     #TEMP2,TEMP1    ; PLACE THE ADDRESS OF TEMP2 IN LOCATION TEMP1
013144  012704  000126                  MOV     #ADR,R4
013150  012714  000130                  MOV     #ADR1,(R4)      ; PLACE THE ADDRESS OF ADR1 IN ADR POINTED BY R4
013154  012734  125252                  MOV     #125252,@(R4)+  ; PLACE THE #125252 IN LOCATION ADR1
013160  057432  177776                  BIS     @-2(R4),@(R2)+  ; SET EVERY OTHER BIT AT LOCATION TEMP2
                                                                ; AND INCREMENT R2 BY 2
013164  010200                          MOV     R2,R0           ; PLACE ADDRESS OF TEMP2 IN R0
013166  025027  177777                  CMP     @-(R0),#177777  ; TEMP2 SHOULD CONTAIN ALL 1'S
013172  001404                          BEQ     4$
013174  012745  000274                  MOV     #274,-(R5)
013200  005245                          INC     -(R5)
013202  000000                          HALT                    ; CMP OR BIS INSTUCTIONS FAILED IN MODES OTHER THAN 0
013204  020227  000142          4$:     CMP     R2,#TEMP1+2     ; R2 SHOULD CONTAIN THE ADDRESS FOR TEMP2
                                                                ; I.E. TEMP1+2
013210  001404                          BEQ     5$
013212  012745  000275                  MOV     #275,-(R5)
013216  005245                          INC     -(R5)
013220  000000                          HALT                    ; MODE 5 IS FAILING
013222  005040                  5$:     CLR     -(R0)           ; PLACE A 0 IN LOCATION TEMP
013224  010067  164712                  MOV     R0,TEMP2        ; PLACE ADDRESS OF TEMP IN LOCATION TEMP2
013230  022020                          CMP     (R0)+,(R0)+     ; BUMP R0 BY 4
013232  055070  000002                  BIS     @-(R0),@2(R0)   ; PLACE THE CONTENTS OF LOCATION TEMP2 AT TEMP
013236  022767  000136  164672          CMP     #TEMP,TEMP      ; LOCATION TEMP SHOULD CONTAIN ITS OWN ADDRESS
013244  001404                          BEQ     BIC1
013246                          ECMP1:
013246  012745  000276                  MOV     #276,-(R5)
```

```
013252  005245                          INC     -(R5)
013254  000000                          HALT                    ; CMP OR BIS INSTRUCTIONS FAILED OR WRONG
                                                                ; SEQUENCE COUNTER


                                ;;*******************************************************************************
                                ;*TEST: 101      NEW INSTRUCTIONS USED IN THIS SECTION ARE BIC, BIT
                                ;;*******************************************************************************

013256                          BIC1:
013256  021527  000101                  CMP     (R5),#101
013262  001122                          BNE     EBIC1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
013264  005215                          INC     (R5)
013266  012703  000136                  MOV     #TEMP,R3        ; LOAD ADDRESS
013272  012713  177777                  MOV     #177777,(R3)    ; LOAD LOCATION
013276  012704  000126                  MOV     #ADR,R4         ; PLACE THE ADDRESS OF ADR IN R4
013302  012714  000130                  MOV     #ADR1,(R4)      ; PLACE THE ADDRESS OF ADR1 IN ADR
013306  011334                          MOV     (R3),@(R4)+     ; LOAD LOCATION ADR1 WITH #177777
013310  012700  000140                  MOV     #TEMP1,R0       ; PLACE THE ADDRESS OF TEMP1 IN R0
013314  012710  125252                  MOV     #125252,(R0)    ; SET EVERY OTHER BIT AT LOCATION TEMP1
013320  000277                          SCC
013322  042013                          BIC     (R0)+,(R3)      ; CLEAR EVERY OTHER BIT
013324  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
013330  034013                          BIT     -(R0),(R3)      ; CHECK IT
013332  001404                          BEQ     1$              ; CONTINUE IF OK
013334  012745  000277                  MOV     #277,-(R5)
013340  005245                          INC     -(R5)
013342  000000                          HALT                    ; BIC OR BIT INSTRUCTION FAILED
013344  032713  052525          1$:     BIT     #52525,(R3)     ; CHECK IT
013350  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
013354  056013  000000                  BIS     0(R0),(R3)      ; SET THE BITS THAT WERE CLEARED
013360  100404                          BMI     2$              ; CONTINUE IF OK
013362  012745  000300                  MOV     #300,-(R5)
013366  005245                          INC     -(R5)
013370  000000                          HALT                    ; BIT OR BIS INSTRUCTION FAILED
013372  012720  077777          2$:     MOV     #77777,(R0)+    ; SET ALL THE BITS AT LOCATION TEMP1 EXCEPT SIGN BIT
013376  010002                          MOV     R0,R2
013400  046213  177776                  BIC     -2(R2),(R3)     ; TRY CLEARING THE OTHER BITS
013404  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
013410  020027  000142                  CMP     R0,#TEMP1+2     ; R0 SHOULD CONTAIN THE ADDRESS OF TEMP1+2
013414  001404                          BEQ     3$
013416  012745  000301                  MOV     #301,-(R5)
013422  005245                          INC     -(R5)
013424  000000                          HALT
013426  010020                  3$:     MOV     R0,(R0)+        ; PLACE THE ADDRES OF LOCATION TEMP2 IN TEMP2
013430  000263                          SEVC                    ; SET V & C BITS
013432  045000                          BIC     @-(R0),R0       ; CLEAR R0
013434  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
013440  037413  177776                  BIT     @-2(R4),(R3)    ; CHECK IT
013444  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
013450  012746  125252                  MOV     #125252,-(SP)   ; SET EVERY OTHER BIT ON THE STACK
013454  017423  177776                  MOV     @-2(R4),(R3)+   ; SET ALL THE BITS AT LOCATION TEMP
013460  046643  000000                  BIC     0(SP),-(R3)     ; CLEAR EVERY OTHER BIT AT LOCATION TEMP
013464  022327  052525                  CMP     (R3)+,#52525    ; TEMP SHOULD CONTAIN # 52525
013470  001404                          BEQ     4$
```

```
013472  012745  000302                  MOV     #302,-(R5)
013476  005245                          INC     -(R5)
013500  000000                          HALT                        ; BIC FAILED IN MODE 6
013502  012700  000144          4$:     MOV     #TEMP2+2,R0         ; PLACE THE ADDRESS OF TEMP2+2 IN R0
013506  010340                          MOV     R3,-(R0)            ; PLACE THE ADDRESS OF TEMP1 IN TEMP2
013510  014330                          MOV     -(R3),@(R0)+        ; MOVE # 52525 IN LOCATION TEMP1
013512  000263                          SEVC                        ; SET V & C BITS
013514  035026                          BIT     @-(R0),(SP)+        ; BIT TEST TEMP1 WITH STACK AND RESTORE STACK POINTER
013516  004737  017172                  JSR     PC,@#$CC5           ; CHECK FOR CC = 5
013522  020627  000450                  CMP     SP,#START           ; MAKE SURE THAT THE SP IS OK
013526  001404                          BEQ     INC1
013530                          EBIC1:
013530  012745  000303                  MOV     #303,-(R5)
013534  005245                          INC     -(R5)
013536  000000                          HALT                        ; STACK POINTER FOULED UP OR SEQUENCE ERROR
```

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

013540                          INC1:
013540  021527  000102                  CMP     (R5),#102
013544  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
013546  012745  000304                  MOV     #304,-(R5)
013552  005245                          INC     -(R5)
013554  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
013556  005215                  2$:     INC     (R5)
013560  012704  000140                  MOV     #TEMP1,R4       ; LOAD ADDRESS
013564  012714  077777                  MOV     #77777,(R4)     ; TEMP1 = 77777
013570  000261                          SEC
013572  005214                          INC     (R4)            ; ADD ONES INTO LOCATION
013574  004737  017320                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
013600  012714  177776                  MOV     #177776,(R4)
013604  012700  000136                  MOV     #TEMP,R0        ; R0 IS POINTING TO LOCATION TEMP
013610  012710  017256                  MOV     #$CC11,(R0)     ; PLACE THE ADDRESS OF SUBROUTINE TO CHECK CC = 11
                                                                ; IN LOCATION TEMP
013614  005214                          INC     (R4)            ;
013616  004730                          JSR     PC,@(R0)+       ; CHECK FOR CC = 11
013620  005214                          INC     (R4)            ;
013622  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
013626  005214                          INC     (R4)
013630  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1
013634  026427  000000  000001          CMP     0(R4),#1        ; CHECK IT
013642  001404                          BEQ     4$              ; CONTINUE IF OK
013644  012745  000305                  MOV     #305,-(R5)
013650  005245                          INC     -(R5)
013652  000000                          HALT                    ; INC INSTRUCTION FAILED
013654  000261                  4$:     SEC
013656  005314                          DEC     (R4)            ; SUBTRACT ONES FROM LOCATION
013660  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC = 5
013664  005314                          DEC     (R4)
013666  004770  177776                  JSR     PC,@-2(R0)      ; CHECK FOR CC = 11
013672  012714  100000                  MOV     #100000,(R4)
013676  005314                          DEC     (R4)            ;
013700  004737  017126                  JSR     PC,@#$CC3       ; CHECK FOR CC = 3
013704  005314                          DEC     (R4)            ;
013706  004737  017066                  JSR     PC,@#$CC1       ; CHECK FOR CC = 1


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

013712                          COM1:
013712  021527  000103                  CMP     (R5),#103
013716  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
013720  012745  000306                  MOV     #306,-(R5)
013724  005245                          INC     -(R5)
013726  000000                          HALT                    ; TEST IS IN WRONG SEQUENCE
013730  005215                  1$:     INC     (R5)
013732  012703  000140                  MOV     #TEMP1,R3       ; LOAD ADDRESS
```

```
013736  012713  125252                MOV     #125252,(R3)     ; LOAD EVERY OTHER BIT
013742  000277                        SCC
013744  005163  000000                COM     0(R3)            ; 1'S COMPLEMENT
013750  004737  017066                JSR     PC,@#$CC1        ; CHECK FOR CC = 1
013754  022713  052525                CMP     #52525,(R3)      ; CHECK IT
013760  001404                        BEQ     2$               ; CONTINUE IF OK
013762  012745  000307                MOV     #307,-(R5)
013766  005245                        INC     -(R5)
013770  000000                        HALT                     ; COM INSTRUCTION FAILED
013772  000277                2$:     SCC
013774  005123                        COM     (R3)+            ; COMPLEMENT BACK
013776  004737  017256                JSR     PC,@#$CC11       ; CHECK FOR CC = 11
014002  022743  125252                CMP     #125252,-(R3)    ; CHECK IT
014006  001404                        BEQ     3$               ; CONTINUE IF OK
014010  012745  000310                MOV     #310,-(R5)
014014  005245                        INC     -(R5)
014016  000000                        HALT                     ; COM INSTRUCTION FAILED
014020  010300                3$:     MOV     R3,R0            ; R0 IS NOW POINTING TO LOCATION TEMP1
014022  012710  177777                MOV     #177777,(R0)
014026  000277                        SCC
014030  005110                        COM     (R0)
014032  004737  017172                JSR     PC,@#$CC5        ; CHECK FOR CC = 5
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 104      NEW INSTRUCTION IN THIS SECTION IS NEG
                                        ;;*******************************************************************************

014036                                  NEG1:
014036  021527  000104                          CMP     (R5),#104
014042  001033                                  BNE     ENEG1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
014044  005215                          1$:     INC     (R5)
014046  012704  000140                          MOV     #TEMP1,R4       ; LOAD ADDRESS
014052  012724  000001                          MOV     #1,(R4)+                ; LOAD THE LOCATION
014056  010402                                  MOV     R4,R2
014060  012762  100000  000000                  MOV     #100000,0(R2)
014066  005444                                  NEG     -(R4)           ; 2'S COMPLEMENT
014070  004737  017256                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
014074  022724  177777                          CMP     #177777,(R4)+   ; CHECK IT
014100  001404                                  BEQ     2$              ; CONTINUE IF OK
014102  012745  000311                          MOV     #311,-(R5)
014106  005245                                  INC     -(R5)
014110  000000                                  HALT                    ; NEG INSTRUCTION FAILED
014112  016444  000000                  2$:     MOV     0(R4),-(R4)     ; TEMP1 CONTAINS THE LARGEST NEGATIVE NUMBER
014116  005414                                  NEG     (R4)            ; 2'S COMPLEMENT
014120  004737  017320                          JSR     PC,@#$CC13      ; CHECK FOR CC = 13
014124  026214  000000                          CMP     0(R2),(R4)      ; CHECK IT
014130  001404                                  BEQ     ROL1            ; CONTINUE IF OK
014132                                  ENEG1:
014132  012745  000312                          MOV     #312,-(R5)
014136  005245                                  INC     -(R5)
014140  000000                                  HALT                    ; WRONG RESULT IN TEMP2 OR WRONG SEQUENCE


                                        ;;*******************************************************************************
                                        ;*TEST: 105      NEW INSTRUCTION IN THIS SECTION IS ROL
                                        ;;*******************************************************************************

014142                                  ROL1:
014142  021527  000105                          CMP     (R5),#105
014146  001032                                  BNE     EROL1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
014150  005215                                  INC     (R5)
014152  012701  000142                          MOV     #TEMP2,R1       ; LOAD ADDRESS
014156  012711  020000                          MOV     #20000,(R1)     ; LOAD LOCATION
014162  000257                                  CCC                     ; CLEAR FLAGS
014164  006121                                  ROL     (R1)+           ; SHIFT
014166  006141                                  ROL     -(R1)           ;
014170  004737  017300                          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
014174  022711  100000                          CMP     #100000,(R1)    ; CHECK IT
014200  001404                                  BEQ     1$              ; CONTINUE IF OK
014202  012745  000313                          MOV     #313,-(R5)
014206  005245                                  INC     -(R5)
014210  000000                                  HALT                    ; ROL INSTRUCTION FAILED
014212  006161  000000                  1$:     ROL     0(R1)           ; SHIFT
014216  004737  017214                          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
014222  010102                                  MOV     R1,R2           ; R2 IS NOW POINTING TO LOCATION TEMP2
014224  006112                                  ROL     (R2)            ; SHIFT
014226  022711  000001                          CMP     #1,(R1)         ; CHECK IT
014232  001404                                  BEQ     ROR1            ; CONTINUE IF OK
```

```
014234                          EROL1:
014234  012745  000314                  MOV     #314,-(R5)
014240  005245                          INC     -(R5)
014242  000000                          HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;*TEST: 106      NEW INSTRUCTION IN THIS SECTION IS ROR
                                ;;*******************************************************************************

014244                          ROR1:
014244  021527  000106                  CMP     (R5),#106
014250  001030                          BNE     EROR1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
014252  005215                          INC     (R5)
014254  012702  000142                  MOV     #TEMP2,R2       ; LOAD ADDRESS
014260  012712  000004                  MOV     #4,(R2)         ; LOAD LOCATION
014264  000257                          CCC                     ; CLEAR FLAGS
014266  006012                          ROR     (R2)            ; SHIFT
014270  006012                          ROR     (R2)            ;
014272  022712  000001                  CMP     #1,(R2)         ; CHECK IT
014276  001404                          BEQ     1$              ; CONTINUE IF OK
014300  012745  000315                  MOV     #315,-(R5)
014304  005245                          INC     -(R5)
014306  000000                          HALT                    ; ROR INSTRUCTION FAILED
014310  006012                  1$:     ROR     (R2)            ; SHIFT
014312  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
014316  006012                          ROR     (R2)            ; SHIFT
014320  004737  017300                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
014324  022712  100000                  CMP     #100000,(R2)    ; CHECK IT
014330  001404                          BEQ     ASL1            ; CONTINUE IF OK
014332                          EROR1:
014332  012745  000316                  MOV     #316,-(R5)
014336  005245                          INC     -(R5)
014340  000000                          HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE


                                ;;*******************************************************************************
                                ;*TEST: 107      NEW INSTRUCTION IN THIS SECTION IS ASL
                                ;;*******************************************************************************

014342                          ASL1:
014342  021527  000107                  CMP     (R5),#107
014346  001404                          BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
014350  012745  000317                  MOV     #317,-(R5)
014354  005245                          INC     -(R5)
014356  000000                          HALT                    ; PROGRAM IS IN WRONG SEQUENCE
014360  005215                  2$:     INC     (R5)
014362  012703  000142                  MOV     #TEMP2,R3       ; LOAD ADDRESS
014366  012713  020000                  MOV     #20000,(R3)     ; LOAD LOCATION
014372  000257                          CCC                     ; CLEAR FLAGS
014374  006313                          ASL     (R3)            ; SHIFT
014376  006313                          ASL     (R3)            ;
014400  004737  017300                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
014404  022713  100000                  CMP     #100000,(R3)    ; CHECK IT
014410  001404                          BEQ     4$              ; CONTINUE IF OK
014412  012745  000320                  MOV     #320,-(R5)
014416  005245                          INC     -(R5)
014420  000000                          HALT                    ; ASL INSTRUCTION FAILED
014422  006313                  4$:     ASL     (R3)            ; SHIFT
014424  004737  017214                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
014430  006313                          ASL     (R3)            ; SHIFT
```

```
3152  014432  004737  017150          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
```

```
                                        ;;*******************************************************************************
                                        ;*TEST: 110     NEW INSTRUCTION IN THIS SECTION IS ASR
                                        ;;*******************************************************************************

014436                                  ASR1:
014436  021527  000110                          CMP     (R5),#110
014442  001040                                  BNE     EASR1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
014444  005215                          1$:     INC     (R5)
014446  012704  000142                          MOV     #TEMP2,R4       ; LOAD ADDRESSES
014452  012703  000136                          MOV     #TEMP,R3        ;
014456  012714  000004                          MOV     #4,(R4)         ; LOAD LOCATION
014462  000257                                  CCC                     ; CLEAR FLAGS
014464  006214                                  ASR     (R4)            ; SHIFT
014466  006214                                  ASR     (R4)            ;
014470  022714  000001                          CMP     #1,(R4)         ; CHECK IT
014474  001404                                  BEQ     2$              ; CONTINUE IF OK
014476  012745  000321                          MOV     #321,-(R5)
014502  005245                                  INC     -(R5)
014504  000000                                  HALT                    ; ASR INSTRUCTION FAILED
014506  006214                          2$:     ASR     (R4)            ; SHIFT
014510  004737  017214                          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
014514  006214                                  ASR     (R4)            ; SHIFT
014516  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
014522  012713  100002                          MOV     #100002,(R3)    ; LOAD LOCATION
014526  006213                                  ASR     (R3)            ; SHIFT
014530  006213                                  ASR     (R3)            ;
014532  004737  017256                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
014536  022713  160000                          CMP     #160000,(R3)    ; CHECK IT
014542  001404                                  BEQ     ADC1            ; CONTINUE IF OK
014544                                  EASR1:
014544  012745  000322                          MOV     #322,-(R5)
014550  005245                                  INC     -(R5)
014552  000000                                  HALT                    ; WRONG RESULT IN TEMP OR WRONG SEQUENCE


                                        ;;*******************************************************************************
                                        ;*TEST: 111     NEW INSTRUCTION IN THIS SECTION IS ADC
                                        ;;*******************************************************************************

014554                                  ADC1:
014554  021527  000111                          CMP     (R5),#111
014560  001404                                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
014562  012745  000323                          MOV     #323,-(R5)
014566  005245                                  INC     -(R5)
014570  000000                                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
014572  005215                          2$:     INC     (R5)
014574  012700  000136                          MOV     #TEMP,R0        ; LOAD ADDRESS
014600  005010                                  CLR     (R0)            ; CLEAR THE LOCATION
014602  000257                                  CCC                     ; CLEAR FLAGS
014604  005510                                  ADC     (R0)            ; ADD C BIT = 0
014606  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
014612  000261                                  SEC                     ; C=1
014614  005510                                  ADC     (R0)            ; ADD C BIT=1
014616  000261                                  SEC                     ; C=1
014620  005510                                  ADC     (R0)            ; AGAIN
```

```
014622  004737  017046                JSR     PC,@#$CC0       ; CHECK FOR CC = 0
014626  022710  000002                CMP     #2,(R0)         ; CHECK IT
014632  001404                        BEQ     4$              ; CONTINUE IF OK
014634  012745  000324                MOV     #324,-(R5)
014640  005245                        INC     -(R5)
014642  000000                        HALT                    ; ADC INSTRUCTION FAILED
014644  012710  077777          4$:   MOV     #77777,(R0)     ; LOAD LARGEST POSITVE NUMBER
014650  000261                        SEC                     ; C=1
014652  005510                        ADC     (R0)            ; ADD C BIT=1
014654  004737  017300                JSR     PC,@#$CC12      ; CHECK FOR CC = 12
014660  022710  100000                CMP     #100000,(R0)    ; CHECK IT
014664  001404                        BEQ     6$              ; CONTINUE IF OK
014666  012745  000325                MOV     #325,-(R5)
014672  005245                        INC     -(R5)
014674  000000                        HALT                    ; ADC INSTRUCTION AILED
014676  012710  177777          6$:   MOV     #-1,(R0)        ; LOAD -1
014702  000261                        SEC                     ; C=1
014704  005510                        ADC     (R0)            ; ADD C BIT=1
014706  004737  017172                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

```
                                ;;*******************************************************************
                                ;*TEST: 112     NEW INSTRUCTION IN THIS SECTION IS SBC
                                ;;*******************************************************************

014712                          SBC1:
014712  021527  000112                  CMP     (R5),#112
014716  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
014720  012745  000326                  MOV     #326,-(R5)
014724  005245                          INC     -(R5)
014726  000000                          HALT                    ; TEST IS IN WRONG SEQUENCE
014730  005215                  1$:     INC     (R5)
014732  012701  000136                  MOV     #TEMP,R1        ; LOAD ADDRESS
014736  012711  000003                  MOV     #3,(R1)         ; LOAD LOCATION
014742  000257                          CCC                     ; CLEAR FLAGS
014744  005611                          SBC     (R1)            ; SUBTRACT C BIT=0
014746  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
014752  022711  000003                  CMP     #3,(R1)         ; CHECK IT
014756  001404                          BEQ     2$              ; CONTINUE IF OK
014760  012745  000327                  MOV     #327,-(R5)
014764  005245                          INC     -(R5)
014766  000000                          HALT                    ; SBC INSTRUCTION FAILED
014770  000261                  2$:     SEC                     ; C=1
014772  005611                          SBC     (R1)            ; SUBTRACT C BIT=1
014774  000261                          SEC                     ; C=1
014776  005611                          SBC     (R1)            ;
015000  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
015004  022711  000001                  CMP     #1,(R1)         ; CHECK IT
015010  001404                          BEQ     3$
015012  012745  000330                  MOV     #330,-(R5)
015016  005245                          INC     -(R5)
015020  000000                          HALT                    ; SBC INSTRUCTION FAILED
015022  000261                  3$:     SEC                     ; C=1
015024  005611                          SBC     (R1)            ; SUBTRACT C BIT=1
015026  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
015032  000261                          SEC                     ; C=1
015034  005611                          SBC     (R1)            ; SUBTRACT C BIT = 1
015036  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
015042  022711  177777                  CMP     #-1,(R1)        ; CHECK IT
015046  001404                          BEQ     4$              ; CONTINUE IF OK
015050  012745  000331                  MOV     #331,-(R5)
015054  005245                          INC     -(R5)
015056  000000                          HALT                    ; SBC INSTRUCTION FAILED
015060  012711  100000          4$:     MOV     #100000,(R1)    ; LOAD R1
015064  000261                          SEC                     ; C=1
015066  005611                          SBC     (R1)            ; SUBTRACT C BIT = 1
015070  004737  017106                  JSR     PC,@#$CC2       ; CHECK FOR CC = 2


                                ;;*******************************************************************
                                ;*TEST: 113     NEW INSTRUCTION IN THIS SECTION IS SXT
                                ;;*******************************************************************

015074                          SXT1:
015074  021527  000113                  CMP     (R5),#113
015100  001026                          BNE     ESXT1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
```

```
015102  005215                  1$:     INC     (R5)
015104  012702  000140                  MOV     #TEMP1,R2           ; LOAD ADDRESS
015110  005012                          CLR     (R2)                ; CLEAR LOCATIONS
015112  000277                          SCC
015114  000254                          CLNZ
015116  006712                          SXT     (R2)                ; SIGN EXTEND
015120  004737  017172                  JSR     PC,@#$CC5           ; CHECK FOR CC = 5
015124  005712                          TST     (R2)                ; LOCATION SHOULD STILL BE 0
015126  001404                          BEQ     2$                  ; CONTINUE IF OK
015130  012745  000332                  MOV     #332,-(R5)
015134  005245                          INC     -(R5)
015136  000000                          HALT                        ; SXT INSTRUCTION FAILED
015140  000273                  2$:     SENVC                       ; SET N, V & C BITS
015142  006712                          SXT     (R2)                ; SIGN EXTEND
015144  004737  017256                  JSR     PC,@#$CC11          ; CHECK FOR CC = 11
015150  022712  177777                  CMP     #-1,(R2)            ; LOCATION SHOULD NEW HAVE -1
015154  001404                          BEQ     SWAB1               ; CONTINUE IF OK
015156                          ESXT1:
015156  012745  000333                  MOV     #333,-(R5)
015162  005245                          INC     -(R5)
015164  000000                          HALT                        ; WRONG RESULT IN TEMP1 OR WRONG SEQUENCE
```

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

015166                                  SWAB1:
015166  021527  000114                          CMP     (R5),#114
015172  001034                                  BNE     ESWAB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
015174  005215                                  INC     (R5)
015176  012703  000142                          MOV     #TEMP2,R3       ; LOAD ADDRESS
015202  012713  125125                          MOV     #125125,(R3)    ; LOAD BIT PATTERN INTO LOCATION
015206  000277                                  SCC
015210  000250                                  CLN
015212  000313                                  SWAB    (R3)            ; SWAP BYTES OF LOCATIONS
015214  004737  017236                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
015220  022713  052652                          CMP     #52652,(R3)     ; CHECK IT
015224  001404                                  BEQ     1$              ; CONTINUE IF OK
015226  012745  000334                          MOV     #334,-(R5)
015232  005245                                  INC     -(R5)
015234  000000                                  HALT                    ; SWAB INSTRUCTION FAILED
015236  012713  000377                  1$:     MOV     #377,(R3)
015242  000277                                  SCC
015244  000244                                  CLZ
015246  000363  000000                          SWAB    0(R3)
015252  004737  017150                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
015256  022713  177400                          CMP     #177400,(R3)
015262  001404                                  BEQ     XOR1
015264                                  ESWAB1:
015264  012745  000335                          MOV     #335,-(R5)
015270  005245                                  INC     -(R5)
015272  000000                                  HALT                    ; WRONG RESULT IN: TEMP2 OR WRONG SEQUENCE


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

015274                                  XOR1:
015274  021527  000115                          CMP     (R5),#115
015300  001041                                  BNE     EXOR1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
015302  005215                                  INC     (R5)
015304  012704  177777                          MOV     #-1,R4          ; LOAD LOCATIONS
015310  012767  177777  162622                  MOV     #-1,TEMP1       ;
015316  000277                                  SCC
015320  074467  162614                          XOR     R4,TEMP1        ; SHOULD PRODUCE 0'S IN TEMP1
015324  004737  017172                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
015330  012767  077777  162602                  MOV     #77777,TEMP1
015336  012700  000140                          MOV     #TEMP1,R0       ; PLACE THE ADDRESS OF TEMP1 IN R0
015342  000263                                  SEVC                    ; SET V & C BITS
015344  000244                                  CLZ
015346  074410                                  XOR     R4,(R0)
015350  004737  017256                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
015354  012701  125252                          MOV     #125252,R1      ; LOAD LOCATIONS
015360  012720  052525                          MOV     #52525,(R0)+    ;
015364  000277                                  SCC
015366  074140                                  XOR     R1,-(R0)        ; SHOULD PRODUCE ALL 1'S IN TEMP1
```

```
015370  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
015374  022737  177777  000140          CMP     #-1,@#TEMP1     ; CHECK IT
015402  001404                          BEQ     ADD1            ; CONTINUE IF OK
015404                          EXOR1:
015404  012745  000336                  MOV     #336,-(R5)
015410  005245                          INC     -(R5)
015412  000000                          HALT                    ; WRONG RESULT IN TEMP1 OR WRONG SEQUENCE
```

```
                                        ;;*******************************************************************
                                        ;*TEST: 116     NEW INSTRUCTION IN THIS SECTION IS ADD
                                        ;;*******************************************************************

015414                                  ADD1:
015414  021527  000116                          CMP     (R5),#116
015420  001133                                  BNE     EADD1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
015422  005215                                  INC     (R5)
015424  012700  000142                          MOV     #TEMP2,R0       ; LOAD ADDRESSES
015430  012701  000136                          MOV     #TEMP,R1        ;
015434  012767  021421  162500                  MOV     #21421,TEMP2    ; LOAD LOCATIONS
015442  011011                                  MOV     (R0),(R1)       ;
015444  061011                                  ADD     (R0),(R1)       ; ADD
015446  004737  017046                          JSR     PC,@#$CC0       ; CHECK FOR CC = 0
015452  022767  043042  162456                  CMP     #43042,TEMP     ; CHECK IT
015460  001404                                  BEQ     1$              ; CONTINUE IF OK
015462  012745  000337                          MOV     #337,-(R5)
015466  005245                                  INC     -(R5)
015470  000000                                  HALT                    ; ADD INSTRUCTION FAILED
015472  005010                          1$:     CLR     (R0)            ; CLEAR LOCATION TEMP2
015474  060020                                  ADD     R0,(R0)+        ; PLACE THE ADDRESS OF TEMP2 IN LOCATION TEMP2
015476  024027  000142                          CMP     -(R0),#TEMP2    ; CHECK IT
015502  001404                                  BEQ     2$
015504  012745  000340                          MOV     #340,-(R5)
015510  005245                                  INC     -(R5)
015512  000000                                  HALT                    ; ADD INSTRUCTION FAILED IN MODE 2
015514  012767  156357  162420          2$:     MOV     #-21421,TEMP2   ; LOAD LOCATIONS
015522  012011                                  MOV     (R0)+,(R1)      ;
015524  064011                                  ADD     -(R0),(R1)      ; ADD
015526  004737  017256                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
015532  022767  134736  162376                  CMP     #-43042,TEMP    ; CHECK IT
015540  001404                                  BEQ     3$              ;CONTINUE IF OK
015542  012745  000341                          MOV     #341,-(R5)
015546  005245                                  INC     -(R5)
015550  000000                                  HALT                    ; ADD INSTRUCTION FAILED
015552  012767  100000  162362          3$:     MOV     #100000,TEMP2   ; LOAD LOCATIONS
015560  011061  000000                          MOV     (R0),0(R1)      ;
015564  066011  000000                          ADD     0(R0),(R1)      ; ADD SHOULD RESULT AS 0'S
015570  004737  017214                          JSR     PC,@#$CC7       ; CHECK FOR CC=7
015574  012767  021421  162336                  MOV     #21421,TEMP1    ; LOAD LOCATION TEMP1
015602  012760  000140  000000                  MOV     #TEMP1,0(R0)    ; PLACE THE ADDRESS OF LOCATION TEMP1 IN TEMP2
015610  012711  156357                          MOV     #-21421,(R1)    ; LOAD LOCATION TEMP
015614  010004                                  MOV     R0,R4           ; MAKE R4 POINT TO LOCATION TEMP2
015616  067411  000000                          ADD     @0(R4),(R1)     ; ADD SHOULD RESULT AS 0'S
015622  004737  017172                          JSR     PC,@#$CC5       ; CHECK FOR CC=5
015626  005430                                  NEG     @(R0)+          ; NEGATE THE CONTENTS OF TEMP1
015630  012746  021421                          MOV     #21421,-(SP)    ; PLACE # 21421 ON THE STACK
015634  065066  000000                          ADD     @-(R0),0(SP)    ; ADD, SHOULD=0'S
015640  004737  017172                          JSR     PC,@#$CC5       ; CHECK FOR CC=5
015644  005726                                  TST     (SP)+           ; CHECK THE STACK TO CONTAIN 0, ALSO
                                                                        ; RESTORE THE STACK POINTER
015646  001404                                  BEQ     4$
015650  012745  000342                          MOV     #342,-(R5)
015654  005245                                  INC     -(R5)
015656  000000                                  HALT                    ; ADD INSTRUCTION FAILED IN MODE 5
015660  012767  137777  162254          4$:     MOV     #137777,TEMP2
```

```
015666  062767  137777  162246          ADD     #137777,TEMP2
015674  004737  017126                  JSR     PC,@#$CC3       ; CHECK CC=3
015700  022767  077776  162234          CMP     #77776,TEMP2
015706  001404                          BEQ     SUB1
015710                          EADD1:
015710  012745  000343                  MOV     #343,-(R5)
015714  005245                          INC     -(R5)
015716  000000                          HALT                    ; WRONG RESULT AT TEMP OR WRONG SEQUENCE


                                ;;*******************************************************************
                                ;*TEST: 117      NEW INSTRUCTION IN THIS SECTION IS SUB
                                ;;*******************************************************************

015720                          SUB1:
015720  021527  000117                  CMP     (R5),#117
015724  001100                          BNE     ESUB1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
015726  005215                          INC     (R5)
015730  012702  000136                  MOV     #TEMP,R2        ; LOAD ADDRESSES
015734  012703  000140                  MOV     #TEMP1,R3       ;
015740  012767  021421  162170          MOV     #21421,TEMP     ; LOAD LOCATIONS
015746  012767  156357  162164          MOV     #-21421,TEMP1   ;
015754  161213                          SUB     (R2),(R3)       ; RESULT SHOULD=-43042
015756  004737  017236                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
015762  022767  134736  162150          CMP     #-43042,TEMP1   ; CHECK IT
015770  001404                          BEQ     1$              ;CONTINUE IF OK
015772  012745  000344                  MOV     #344,-(R5)
015776  005245                          INC     -(R5)
016000  000000                          HALT                    ; SUB INSTRUCTION FAILED
016002  012767  021421  162130  1$:     MOV     #21421,TEMP1    ; LOAD LOCATION
016010  161213                          SUB     (R2),(R3)       ; RESULT SHOULD=0
016012  001404                          BEQ     2$
016014  012745  000345                  MOV     #345,-(R5)
016020  005245                          INC     -(R5)
016022  000000                          HALT                    ; SUB INSTRUCTION FAILED
016024  012767  177777  162106  2$:     MOV     #-1,TEMP1       ; LOAD LOCATIONS
016032  012767  077777  162076          MOV     #77777,TEMP     ; LOAD LOCATIONS
016040  161312                          SUB     (R3),(R2)       ; RESULT SHOULD GIVE 100000 AND OVERFLOW
016042  004737  017320                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
016046  022767  100000  162062          CMP     #100000,TEMP    ; CHECK IT
016054  001404                          BEQ     3$              ; CONTINUE IF OK
016056  012745  000346                  MOV     #346,-(R5)
016062  005245                          INC     -(R5)
016064  000000                          HALT                    ; SUB INSTRUCTION FAILED
016066  012712  177777          3$:     MOV     #-1,(R2)
016072  161312                          SUB     (R3),(R2)
016074  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
016100  012767  077777  162030          MOV     #77777,TEMP
016106  162767  077777  162022          SUB     #77777,TEMP
016114  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC=4
016120  005767  162012                  TST     TEMP
016124  001404                          BEQ     SOB             ; TEMP SHOULD BE =0
016126                          ESUB1:
016126  012745  000347                  MOV     #347,-(R5)
016132  005245                          INC     -(R5)
```

```
3480  016134  000000                HALT                    ; SUB INSTRUCTION FAILED OR SEQUENCE ERROR
```

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

016136                          SOB:
016136  021527  000120                  CMP     (R5),#120
016142  001042                          BNE     ESOB            ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
016144  005215                          INC     (R5)
016146  012700  000012                  MOV     #10.,R0         ; LOAD REGISTERS
016152  005001                          CLR     R1              ;
016154  005201                  1$:     INC     R1              ; KEEP COUNT
016156  020127  000012                  CMP     R1,#10.
016162  003404                          BLE     2$
016164  012745  000350                  MOV     #350,-(R5)
016170  005245                          INC     -(R5)
016172  000000                          HALT                    ; SOB INSTRUCTION FAILED
016174  000277                  2$:     SCC
016176  077012                          SOB     R0,1$           ; SUB. 1 FROM REG. 0, GO BACK TO 1$
016200  004737  017340                  JSR     PC,@#$CC17      ; CHECK FOR CC = 17
016204  005700                          TST     R0              ; REG. 0 = 0 ?
016206  001404                          BEQ     3$              ; NO, FAILED
016210  012745  000351                  MOV     #351,-(R5)
016214  005245                          INC     -(R5)
016216  000000                          HALT                    ; SOB INSTRUCTION FAILED
016220  022701  000012          3$:     CMP     #10.,R1         ; DID IT GO THRU 10 TIMES ?
016224  001404                          BEQ     4$              ; CONTINUE IF OK
016226  012745  000352                  MOV     #352,-(R5)
016232  005245                          INC     -(R5)
016234  000000                          HALT                    ; SOB INSTRUCTION FAILED
016236  012704  000010          4$:     MOV     #10,R4          ; PLACE #10 IN R4
016242  077401                  5$:     SOB     R4,5$           ; STAY HERE UNTILL R4 = 0
016244  005704                          TST     R4
016246  001404                          BEQ     PSWN0           ; CONTINUE IF OK
016250                          ESOB:
016250  012745  000353                  MOV     #353,-(R5)
016254  005245                          INC     -(R5)
016256  000000                          HALT                    ; SOB FAILED OR WRONG SEQUENCE


                                ;;*******************************************************************
                                ;*TEST: 121     NEW INSTRUCTIONS IN THIS SECTION ARE MTPS & MFPS
                                ;;*******************************************************************


016260                          PSWN0:
016260  021527  000121                  CMP     (R5),#121
016264  001042                          BNE     EPSWN0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
016266  005215                          INC     (R5)
016270  012700  000176                  MOV     #TEMP,R0        ; PUT THE ADDRESS OF TEMP IN R0
016274  012701  000140                  MOV     #TEMP1,R1       ; PUT THE ADDRESS OF TEMP1 IN R1
016300  012711  177777                  MOV     #177777,(R1)    ; TEMP1 = 177777
016304  005010                          CLR     (R0)            ; TEMP  = 0
016306                                  MTPS    (R0)            ; PSW = 0
016306  106410                          .WORD   106400!..C
016310  004737  017046                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
```

```
016314                                  MFPS    (R1)            ; MOVE PSW TO TEMP1
016314  106711                          .WORD   106700!..C
016316  004737  017150                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
016322  022711  177400                  CMP     #177400,(R1)    ; CHECK TEMP1 TO MAKE SURE THAT  ONLY
                                                                ; THE LOWER BYTE WAS AFFECTED BY MFPS
016326  001404                          BEQ     1$
016330  012745  000354                  MOV     #354,-(R5)
016334  005245                          INC     -(R5)
016336  000000                          HALT                    ; MTPS OR MFPS INSTRUCTION FAILED
016340  005011                  1$:     CLR     (R1)
016342                                  MTPS    #377            ; SET PSW = 357 SINCE T BIT CAN NOT BE SET BY MTPS
016342  106427                          .WORD   106400!..C
016346  004737  017340                  JSR     PC,@#$CC17      ; CHECK FOR CC = 17
016352                                  MFPS    TEMP1           ; MOVE PSW TO TEMP1
016352  106767                          .WORD   106700!..C
016356  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11 [C BIT SHOULD NOT BF EFFECTED BY MFP
016362  022767  000357  161550          CMP     #357,TEMP1
016370  001404                          BEQ     BTWRD
016372                          EPSWN0:
016372  012745  000355                  MOV     #355,-(R5)
016376  005245                          INC     -(R5)
016400  000000                          HALT                    ; MFPS INSTRUCTION FAILED IN MODE 6
                                                                ; OR SEQUENCE ERROR
```

```
                                ;;*******************************************************************
                                ;*TEST: 122     BYTE INSTRUCTIONS REQUIRING WORD INST. TO CHECK
                                ;;*******************************************************************
                                ;;

016402                          BTWRD:
016402  021527  000122                  CMP     (R5),#122
016406  001124                          BNE     EBTWRD          ; IF IN WRONG SEQUNCE GO TO HALT AT THE END OF THE TEST
016410  005215                          INC     (R5)
016412  005000                          CLR     R0
016414  000277                          SCC
016416  112700  000200                  MOVB    #200,R0         ; SET THE HIGHEST BIT OF THE
                                                                ; LOWER BYTE
016422  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC=11
016426  022700  177600                  CMP     #177600,R0      ; CHECK FOR SIGN EXTENSION IN R0
016432  001404                          BEQ     1$
016434  012745  000356                  MOV     #356,-(R5)
016440  005245                          INC     -(R5)
016442  000000                          HALT                    ; SIGN WAS NOT EXTENDED IN R0
016444  000277                  1$:     SCC
016446  012700  177777                  MOV     #177777,R0
016452  112700  000000                  MOVB    #0,R0           ;CLEAR THE LOWER BYTE OF R0.
016456  004737  017172                  JSR     PC,@#$CC5       ; CHECK FOR CC=5
016462  005700                          TST     R0              ; CHECK R0 FOR SIGN EXTENTION
016464  001404                          BEQ     2$
016466  012745  000357                  MOV     #357,-(R5)
016472  005245                          INC     -(R5)
016474  000000                          HALT                    ; SIGN WAS NOT EXTENDED IN R0.
016476  012704  000142          2$:     MOV     #TEMP2,R4       ; R4 IS POINTING TO TEMP2
016502  012714  000377                  MOV     #377,(R4)       ; PLACE #377 IN LOCATION TEMP2
016506  012706  000446                  MOV     #START-2,R6
016512  116426  000000                  MOVB    0(R4),(R6)+     ; PUSH # 377 ON STACK
016516  022706  000450                  CMP     #START,R6
016522  001404                          BEQ     3$
016524  012745  000360                  MOV     #360,-(R5)
016530  005245                          INC     -(R5)
016532  000000                          HALT                    ; R6 DID NOT GET INCREMENTED
                                                                ; BY 2 BY A BYTE INSTRUCTION
016534  124627  000377          3$:     CMPB    -(R6),#377      ; CHECK LOCATION START-2 TO
                                                                ; CONTAIN PROPER DATA
016540  001404                          BEQ     4$
016542  012745  000361                  MOV     #361,-(R5)
016546  005245                          INC     -(R5)
016550  000000                          HALT                    ; BYTE INSTRUCTION IS FAILING WITH R6
016552  022706  000446          4$:     CMP     #START-2,R6     ; CHECK THAT R6 WAS DECREMENTED
                                                                ; BY 2 BY A BYTE INSTRUCTION
016556  001404                          BEQ     5$
016560  012745  000362                  MOV     #362,-(R5)
016564  005245                          INC     -(R5)
016566  000000                          HALT                    ; R6 WAS NOT DECREMENTED
016570  016467  000000  161340  5$:     MOV     0(R4),TEMP      ; SET THE LOWER BYTE OF LOCATION TEMP
016576  005726                          TST     (R6)+           ; RESTORE STACK POINTER
016600  000277                          SCC
016602  114667  161331                  MOVB    -(SP),TEMP+1    ; SET THE HIGHER BYTE OF LOCATION TEMP
016606  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC=11
016612  022767  177777  161316          CMP     #177777,TEMP    ; CHECK TEMP FOR THE CORRECT VALUE
016620  001404                          BEQ     6$
```

```
016622  012745  000363                  MOV     #363,-(R5)
016626  005245                          INC     -(R5)
016630  000000                          HALT                    ; TEMP FOULED UP
016632  005067  161300          6$:     CLR     TEMP
016636  000241                          CLC
016640  105167  161273                  COMB    TEMP+1          ; WRITE 1'S IN THE HIGHER BYTE OF TEMP
016644  004737  017256                  JSR     PC,@#$CC11      ; CHECK FOR CC=11
016650  022767  177400  161260          CMP     #177400,TEMP
016656  001404                          BEQ     NEXT
016660                          EBTWRD:
016660  012745  000364                  MOV     #364,-(R5)
016664  005245                          INC     -(R5)
016666  000000                          HALT                    ; WRONG VALUE IN TEMP OR WRONG SEQUENCE
```

```
                                        ;               END OF PASS
                                        ;               ***********

016670                                  NEXT:
016670  021527  000123                          CMP     (R5),#123
016674  001404                                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
016676  012745  000365                          MOV     #365,-(R5)
016702  005245                                  INC     -(R5)
016704  000000                                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
016706  005267  161172                  2$:     INC     $PASS
016712  126727  161166  000001                  CMPB    $PASS,#1        ; ALLOW THE TYPE OUT OF END OF
                                                                        ; PASS EVERY 377 PASSES
016720  001011                                  BNE     DOAGN
016722  000004  000152                          .TYPE   ,ENDPAS         ; TYPE END OF PASS MESSAGE
016726  013700  000042                  GET42:  MOV     @#42,R0
016732  001404                                  BEQ     DOAGN
016734  004710                          $ENDAD: JSR     PC,(R0)
016736  000240                                  NOP
016740  000240                                  NOP
016742  000240                                  NOP
016744  005067  161132                  DOAGN:  CLR     $TESTN          ; PREPARE TO START FROM TEST 0
016750  000167  161474                  RETURN: JMP     START           ; START TEST OVER AT BEGINNING
                                        ;       ------------------------

                                        ;;*******************************************************************************

                                        .SBTTL  POWER FAIL ROUTINE

016754  012737  016764  000024          PWRDN:  MOV     #PWRUP,@#24     ; GO TO POWER UP ROUTINE AFTER THE POWER COMES BACK
016762  000000                                  HALT

016764  012706  000450                  PWRUP:  MOV     #START,SP
016770  012737  016754  000024                  MOV     #PWRDN,@#24
016776  000004  000166                          .TYPE   ,POWER
017002  000760                                  BR      DOAGN
```

```
                                        .SBTTL  TYPE ROUTINE

017004  132737  000040  000117  TYPE:   BITB    #40,@#$ENVM     ; HAS THE CONSOLE OUTPUTS BEEN SUPPRESSED?
017012  001012                          BNE     4$              ; IF SO THEN GO TO 4$
0170 -  017603  000000                  MOV     @(SP),R3        ; GET ADDRESS OF MESSAGE

017020  105713          1$:             TSTB    (R3)            ; END OF MESSAGE ?
017022  001406                          BEQ     4$              ; YES, GO WRAP IT UP

017024  105777  161114  3$:             TSTB    @TPS            ; READY FOR NEXT CHARACTER ?
017030  100375                          BPL     3$              ; NO, WAIT
017032  112377  161110                  MOVB    (R3)+,@TPB      ; LOAD AND TYPE THE CHARACTER
017036  000770                          BR      1$              ; YES, GET THE NEXT CHARACTER

017040  062716  000002  4$:             ADD     #2,(SP)         ; ADJUST THE RETURN PC
017044  000006                          RTT                     ; RETURN
```

```
017046  003402                  $CC0:   BLE     1$
017050  100401                          BMI     1$
017052  103004                          BCC     2$
017054                          1$:
017054  012745  000366                  MOV     #366,-(R5)
017060  005245                          INC     -(R5)
017062  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 0
017064  000207                  2$:     RTS     PC

017066  003402                  $CC1:   BLE     1$
017070  100401                          BMI     1$
017072  103404                          BCS     2$
017074                          1$:
017074  012745  000367                  MOV     #367,-(R5)
017100  005245                          INC     -(R5)
017102  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 1
017104  000207                  2$:     RTS     PC

017106  100402                  $CC2:   BMI     1$
017110  101401                          BLOS    1$
017112  102404                          BVS     2$
017114                          1$:
017114  012745  000370                  MOV     #370,-(R5)
017120  005245                          INC     -(R5)
017122  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 2
017124  000207                  2$:     RTS     PC

017126  100403                  $CC3:   BMI     1$
017130  001402                          BEQ     1$
017132  102001                          BVC     1$
017134  103404                          BCS     2$
017136                          1$:
017136  012745  000371                  MOV     #371,-(R5)
017142  005245                          INC     -(R5)
017144  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 3
017146  000207                  2$:     RTS     PC

017150  100403                  $CC4:   BMI     1$
017152  001002                          BNE     1$
017154  102401                          BVS     1$
017156  103004                          BCC     2$
017160                          1$:
017160  012745  000372                  MOV     #372,-(R5)
017164  005245                          INC     -(R5)
017166  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 4
017170  000207                  2$:     RTS     PC

017172  100403                  $CC5:   BMI     1$
017174  001002                          BNE     1$
017176  102401                          BVS     1$
017200  103404                          BCS     2$
017202                          1$:
017202  012745  000373                  MOV     #373,-(R5)
017206  005245                          INC     -(R5)
017210  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 5
017212  000207                  2$:     RTS     PC
```

```
017214  100403                  $CC7:   BMI     1$
017216  001002                          BNE     1$
017220  102001                          BVC     1$
017222  103404                          BCS     2$
017224                          1$:
017224  012745  000374                  MOV     #374,-(R5)
017230  005245                          INC     -(R5)
017232  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 7
017234  000207                  2$:     RTS     PC

017236  100002                  $CC10:  BPL     1$
017240  101401                          BLOS    1$
017242  102004                          BVC     2$
017244                          1$:
017244  012745  000375                  MOV     #375,-(R5)
017250  005245                          INC     -(R5)
017252  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 10
017254  000207                  2$:     RTS     PC

017256  100003                  $CC11:  BPL     1$
017260  001402                          BEQ     1$
017262  102401                          BVS     1$
017264  103404                          BCS     2$
017266                          1$:
017266  012745  000376                  MOV     #376,-(R5)
017272  005245                          INC     -(R5)
017274  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 11
017276  000207                  2$:     RTS     PC

017300  100002                  $CC12:  BPL     1$
017302  101401                          BLOS    1$
017304  102404                          BVS     2$
017306                          1$:
017306  012745  000377                  MOV     #377,-(R5)
017312  005245                          INC     -(R5)
017314  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 12
017316  000207                  2$:     RTS     PC

017320  100002                  $CC13:  BPL     1$
017322  003401                          BLE     1$
017324  103404                          BCS     2$
017326                          1$:
017326  012745  000400                  MOV     #400,-(R5)
017332  005245                          INC     -(R5)
017334  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 13
017336  000207                  2$:     RTS     PC

017340  100003                  $CC17:  BPL     1$
017342  001002                          BNE     1$
017344  102001                          BVC     1$
017346  103404                          BCS     2$
017350                          1$:
017350  012745  000401                  MOV     #401,-(R5)
017354  005245                          INC     -(R5)
```

```
017356  000000                  HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 17
017360  000207          2$:     RTS     PC

        000001          .END
```

```
ABASE  = 000000      339
ACDW1  = 000000      339
ACDW2  = 000000      339
ACPUOP= 000000       339     354
ADCB0   004124      1293    1302#
ADCB1   012400      2702    2714#
ADC0    005720      1700    1709#
ADC1    014554      3181    3193#
ADDW0  = 000000      339
ADDW1  = 000000      339
ADDW10= 000000       339
ADDW11= 000000       339
ADDW12= 000000       339
ADDW13= 000000       339
ADDW14= 000000       339
ADDW15= 000000       339
ADDW2  = 000000      339
ADDW3  = 000000      339
ADDW4  = 000000      339
ADDW5  = 000000      339
ADDW6  = 000000      339
ADDW7  = 000000      339
ADDW8  = 000000      339
ADDW9  = 000000      339
ADD0    006524      1875    1887#
ADD1    015414      3363    3372#
ADEVCT= 000000       339     345
ADEVM  = 000000      339
ADR     000126       386#    387    2104*  2107   2124   2219*  2222   2248   2313*  2316   2862   2903
ADR1    000130       388#    389    2105*  2108   2125   2220*  2223   2249   2314*  2317   2863   2904
ADR2    000132       390#    391    2106*  2221*  2224   2250   2315*  2318
AENV   = 000000      339     350
AENVM  = 000000      339     351
AFATAL= 000000       339     342
AMADR1= 000000       339
AMADR2= 000000       339
AMADR3= 000000       339
AMADR4= 000000       339
AMAMS1= 000000       339
AMAMS2= 000000       339
AMAMS3= 000000       339
AMAMS4= 000000       339
AMSGAD= 000000       339     347
AMSGLG= 000000       339     348
AMSGTY= 000000       339     341
AMTYP1= 000000       339
AMTYP2= 000000       339
AMTYP3= 000000       339
AMTYP4= 000000       339
APASS  = 000000      339     344
APRIOR= 000000       339
ASLB0   003726      1236    1245#
ASLB1   012166      2640    2652#
ASL0    005522      1643    1652#
ASL1    014342      3119    3131#
ASRB0   004016      1271#
```

ASRB1	012262	2678#											
ASR0	005612	1678#											
ASR1	014436	3157#											
ASWREG=	000000	339	352										
ATESTN=	000000	339	343										
AUNIT =	000000	339	346										
AUSWR =	000000	339	353										
AVECT1=	000000	339											
AVECT2=	000000	339											
BICB0	003060	1048	1057#										
BICB1	011122	2402	2414#										
BIC0	004630	1446	1455#										
BIC1	013256	2884	2897#										
BRANCH	001110	614	622#										
BTWRD	016402	3554	3564#										
CBIT	000660	516	523#										
CC0	000532	448	451	452	453	454	456	457	458	459	460	461	462	464#
CC1	000576	477	480	481	482	483	484	485	486	488#				
CC2	000646	499	503	504	505	506	507	508	509	510	512#			
CC3	000714	525	530	531	532	533	534	535	537#					
CC4	000764	548	554	555	556	557	558	559	560	562#				
CC5	001076	598	602	604	606	608	610#							
CC6	001032	575	578	579	580	581	583	584	585	586	588#			
CLNZ =	000254	329#	1801	3288										
CMPB0	002720	1017#											
CMPB1	011020	2369	2381#										
CMP0	004536	1428#											
CMP1	013012	2835#											
COMB0	003336	1129#											
COMB1	011544	2509	2527#										
COM0	005132	1535#											
COM1	013712	3006#											
DOAGN	016744	3646	3649	3654#	3669								
DUMMY	000134	392#	393	885	886								
EADD0	006664	1889	1921#										
EADD1	015710	3374	3428#										
EASRB0	004114	1273	1294#										
EASRB1	012370	2680	2703#										
EASR0	005710	1680	1701#										
EASR1	014544	3159	3182#										
EBIC0	004764	1457	1487#										
EBIC1	013530	2899	2954#										
EBTWRD	016660	3566	3625#										
ECMPB0	003050	1019	1049#										
ECMPB1	011112	2383	2403#										
ECMP0	004620	1430	1447#										
ECMP1	013246	2837	2885#										
EINCB1	011534	2472	2510#										
EMODE0	007316	2010	2046#										
EMODE2	007454	2060	2088	2091#									
EMODE3	007646	2102	2130	2133#									
EMODE4	010124	2146	2202	2205#									
EMODE5	010354	2216	2260	2263#									
EMODE6	010524	2276	2299#										
EMODE7	010666	2310	2330#										
ENDCC0	000542	463	468#										

ENDCC1	000606	487	492#											
ENDCC2	000656	511	516#											
ENDCC5	001106	609	614#											
ENDJP1	001276	652	676#											
ENDJP3	001502	688	732#											
ENDJP5	001712	742	786#											
ENDJP7	002112	797	835#											
ENDJSR	002526	845	951#											
ENDPAS	000152	403#	3647											
ENEGB0	003532	1166	1181#											
ENEGB1	011762	2566	2582#											
ENEG0	005326	1572	1587#											
ENEG1	014132	3042	3060#											
EPSW	007132	1972	1995#											
EPSWN0	016372	3528	3555#											
EREGS	002634	962	982#											
EROLB0	003624	1191	1208#											
EROLB1	012060	2595	2613#											
EROL0	005420	1597	1614#											
EROL1	014234	3073	3092#											
ERORB0	003716	1220	1237#											
ERORB1	012156	2623	2641#											
ERORO	005512	1627	1644#											
ERORI	014332	3102	3120#											
ERRNM =	000402	330#	465	468#	489	492#	513	516#	538	541#	563	566#	589	592#
		611	614#	625	628#	630	633#	634	637#	638	641#	642	645#	657
		660#	666	669#	671	674#	677	680#	693	696#	702	705#	707	710#
		712	715#	721	724#	726	729#	733	736#	747	750#	752	755#	757
		760#	762	765#	770	773#	774	777#	779	782#	787	790#	801	804#
		806	809#	811	814#	816	819#	823	826#	829	832#	836	839#	851
		854#	856	859#	865	868#	870	873#	875	878#	881	884#	891	894#
		900	903#	905	908#	910	913#	916	919#	922#	925	928#	931	934#
		939	942#	946	949#	952	955#	983	986#	997	1000#	1026	1029#	1032
		1035#	1037	1040#	1043	1046#	1050	1053#	1060	1063#	1071	1074#	1078	1081#
		1095	1098#	1112	1115#	1132	1135#	1142	1145#	1150	1153#	1173	1176#	1182
		1185#	1200	1203#	1209	1212#	1228	1231#	1238	1241#	1248	1251#	1259	1262#
		1281	1284#	1295	1298#	1305	1308#	1320	1323#	1329	1332#	1346	1349#	1356
		1359#	1366	1369#	1377	1380#	1399	1402#	1415	1418#	1437	1440#	1442	1445#
		1448	1451#	1467	1470#	1474	1477#	1488	1491#	1501	1504#	1518	1521#	1538
		1541#	1548	1551#	1556	1559#	1579	1582#	1588	1591#	1606	1609#	1615	1618#
		1635	1638#	1645	1648#	1655	1658#	1666	1669#	1688	1691#	1702	1705#	1712
		1715#	1727	1730#	1736	1739#	1753	1756#	1763	1766#	1773	1776#	1784	1787#
		1806	1809#	1815	1818#	1836	1839#	1847	1850#	1877	1880#	1896	1899#	1904
		1907#	1914	1917#	1922	1925#	1932	1935#	1942	1945#	1949	1952#	1958	1961#
		1982	1985#	1996	1999#	2017	2020#	2025	2028#	2038	2041#	2047	2050#	2077
		2080#	2092	2095#	2119	2122#	2134	2137#	2155	2158#	2168	2171#	2178	2181#
		2188	2191#	2206	2209#	2234	2237#	2244	2247#	2264	2267#	2291	2294#	2300
		2303#	2323	2326#	2331	2334#	2365	2368#	2371	2374#	2393	2396#	2398	2401#
		2404	2407#	2417	2420#	2431	2434#	2438	2441#	2455	2458#	2460	2463#	2495
		2498#	2511	2514#	2530	2533#	2543	2546#	2552	2555#	2574	2577#	2583	2586#
		2605	2608#	2614	2617#	2632	2635#	2642	2645#	2655	2658#	2667	2670#	2690
		2693#	2704	2707#	2717	2720#	2733	2736#	2742	2745#	2756	2759#	2767	2770#
		2777	2780#	2788	2791#	2811	2814#	2846	2849#	2851	2854#	2870	2873#	2876
		2879#	2886	2889#	2913	2916#	2920	2923#	2929	2932#	2943	2946#	2955	2958#
		2965	2968#	2986	2989#	3009	3012#	3020	3023#	3028	3031#	3052	3055#	3061
		3064#	3083	3086#	3093	3096#	3111	3114#	3121	3124#	3134	3137#	3146	3149#

		3169	3172#	3183	3186#	3196	3199#	3212	3215#	3221	3224#	3235	3238#	3246
		3249#	3256	3259#	3267	3270#	3293	3296#	3302	3305#	3321	3324#	3332	3335#
		3365	3368#	3384	3387#	3391	3394#	3400	3403#	3420	3423#	3429	3432#	3451
		3454#	3457	3460#	3466	3469#	3478	3481#	3494	3497#	3502	3505#	3507	3510#
		3515	3518#	3543	3546#	3556	3559#	3575	3578#	3584	3587#	3593	3596#	3600
		3603#	3606	3609#	3616	3619#	3626	3629#	3640	3643#	3694	3697#	3703	3706#
		3712	3715#	3722	3725#	3732	3735#	3742	3745#	3753	3756#	3762	3765#	3772
		3775#	3781	3784#	3790	3793#	3800	3803#						
ESOB	016250	3487	3514#											
ESUB1	016126	3441	3477#											
ESWAB0	006406	1827	1846#											
ESWAB1	015264	3311	3331#											
ESXT0	006306	1797	1814#											
ESXT1	015156	3283	3301#											
ETSTB1	011010	2347	2370#											
EXOR0	006514	1856	1876#											
EXOR1	015404	3344	3364#											
GET42	016726	3648#												
INCB0	003200	1092#												
INCB1	011350	2470#												
INC0	004774	1486	1498#											
INC1	013540	2953	2962#											
JMP1	001202	645	650#											
JMP2	001306	674	686#											
JMP3	001412	710	716#											
JMP4	001512	729	740#											
JMP5	001610	760	766#											
JMP6	001722	784	795#											
JMP7	002024	814	820#											
JSRM	002264	855	884#											
JSRTST	002122	832	843#											
MARK2	000150	402#	887											
MODE0	007142	1994	2008#											
MODE1	007226	2024	2029#											
MODE2	007326	2045	2058#											
MODE3	007464	2090	2100#											
MODE4	007656	2132	2144#											
MODE5	010134	2204	2214#											
MODE6	010364	2262	2274#											
MODE7	010534	2298	2308#											
NBIT	000544	468	475#											
NEGB0	003452	1164#												
NEGB1	011676	2564#												
NEG0	005246	1570#												
NEG1	014036	3040#												
NEXT	016670	3624	3637#											
NOBIT	000466	437	446#											
NOP1 =	000260	331#	455											
NOTCC	001042	587	596#											
POWER	000166	406#	3668											
PSW	007040	1970#												
PSWN0	016260	3513	3526#											
PWRDN	016754	423	3663#	3667										
PWRUP	016764	3663	3666#											
REGS	002536	924	960#											
RETURN	016750	3655#												

```
ROLB0     003542          1180    1189#
ROLB1     011772          2581    2593#
ROL0      005336          1586    1595#
ROL1      014142          3059    3071#
RORB0     003634          1207    1218#
RORB1     012070          2612    2621#
ROR0      005430          1613    1625#
ROR1      014244          3091    3100#
SBCB0     004256          1343#
SBCB1     012536          2753#
SBC0      006052          1750#
SBC1      014712          3232#
SENVC   = 000273           333#   1809    3296
SEVC    = 000263           332#   1418    1482    1865    2449    2933    2949    3353
SOB       016136          3476    3485#
START     000450           428     434#    847     854     868     908     923    2952    3589    3591    3603    3655    3666
SUB0      006674          1920    1929#
SUB1      015720          3427    3439#
SWAB0     006316          1813    1825#
SWAB1     015166          3300    3309#
SXT0      006230          1795#
SXT1      015074          3281#
TEMP      000136           394#    395     716*    717*    718     729*    730     820     827     832*    833     964*    979
                           981    1460    2029    2062    2069*   2074    2085*   2089    2104    2110*   2112*   2116    2123*
                          2127*   2131    2148*   2149    2166    2173    2181*   2182    2201    2218*   2219    2239    2247*
                          2259    2279    2284    2313    2349    2386    2421    2474    2534    2568    2815    2840    2883
                          2901    2975    3162    3200    3239    3377    3382    3398    3443    3445*   3461*   3464    3472*
                          3473*   3475    3530    3609*   3612*   3614    3619*   3621*   3623
TEMP1     000140           396#    397     766*    767     768*    777     782*    783     784*    2030    2063    2105    2150
                          2183    2220    2280    2294*   2297    2312*   2314    2321    2328    2350    2423    2447    2453
                          2535    2597    2625    2659    2682    2816    2839    2849    2861*   2873    2906    2927    2969
                          3013    3044    3285    3347*   3349*   3351*   3352    3362    3407*   3408    3444    3446*   3449
                          3454*   3460*   3531    3551*   3553
TEMP2     000142           398#    399    2031    2032*   2036    2044    2064    2106    2109*   2114    2122*   2129    2151
                          2153    2175    2184    2186    2203    2221    2241    2261    2278*   2281    2288    2315    2385
                          2683    2721    2760    2860*   2861    2880*   2946    3075    3104    3138    3161    3313    3376
                          3378*   3389    3394*   3403*   3423*   3424*   3426    3587
TPB       000146           401#   3684*
TPS       000144           400#   3682
TSTB0     002644           980     994#
TSTB1     010676          2329    2345#
TST0      004434          1396#
TST1      012720          2808#
TYPE      017004           412    3675#
VBIT      000610           492     497#
XOR0      006416          1845    1854#
XOR1      015274          3330    3342#
YESCC     000774           561     573#
ZBIT      000724           536     546#
$APTHD    000126           372     378#    385
$CC0      017046          1317    1353    1363    1724    1760    1770    1893    1978    2730    2764    2774    3209    3243
                          3253    3381    3536    3690#
$CC1      017066          1068    1075    1109    1124    1139    1464    1471    1515    1530    1545    2428    2435    2492
                          2507    2540    2855    2910    2917    2983    2998    3017    3699#
$CC10     017236          1007    1009    1410    1412    1734    1833    1939    2357    2359    2390    2825    2827    2843
                          2859    3318    3448    3758#
```

```
$CC11    017256    1023   1082   1084   1105   1119   1147   1170   1291   1374   1478   1481   1511   1525
                   1553   1576   1698   1781   1811   1868   1873   1901   1992   2443   2445   2480   2505
                   2548   2571   2700   2785   2926   2937   2976   3025   3049   3179   3264   3298   3356
                   3361   3397   3552   3572   3613   3622   3767#
$CC12    017300    1197   1234   1256   1326   1603   1641   1663   1733   2602   2638   2664   2739   3080
                   3117   3143   3218   3777#
$CC13    017320    1102   1178   1508   1584   1955   2478   2579   2857   2973   3057   3463   3786#
$CC17    017340    1989   3499   3549   3795#
$CC2     017106    1383   1790   2794   3273   3708#
$CC3     017126    1122   1528   2505   2996   3425   3717#
$CC4     017150    1003   1005   1265   1287   1312   1371   1405   1407   1672   1694   1719   1778   1843
                   1963   1985   2353   2355   2465   2488   2673   2696   2725   2782   2819   2821   3152
                   3175   3204   3261   3328   3471   3474   3539   3727#
$CC5     017172    1107   1117   1156   1335   1420   1484   1513   1523   1562   1742   1803   1862   2452
                   2500   2559   2748   2935   2951   2981   2991   3035   3227   3290   3350   3412   3416
                   3581   3737#
$CC7     017214    1204   1232   1263   1285   1610   1639   1670   1692   1909   2609   2636   2671   2694
                   3087   3115   3150   3173   3406   3748#
$CPUOP   000124     354#
$DEVCT   000106     345#
$ENDAD   016734     302    3650#
$ENV     000116     350#
$ENVM    000117     351#   3675
$ETABL   000116     349#    424
$ETEND   000126     361#    384
$FATAL   000100     342#
$HD    = 000003     282     283
$HIBTS   000126     379#
$MAIL    000076     340#    380    384    426
$MBADR   000130     380#
$MSGAD   000112     347#
$MSGLG   000114     348#
$MSGTY   000076     341#
$PASS    000104     344#   3643*  3644
$PASTM   000134     382#
$SVPC  = 001000     300#    305
$SWR   = 160000     282     283#
$SWREG   000120     352#
$TESTN   000102     343#    435    903    977   3654*
$TN    = 000124     282#   334#   442    447    448#   471    476    477#   493    498    499#   519    524
                    525#   542    547    548#   569    574    575#   592    597    598#   618    623    624#
                    646    651    652#   682    687    688#   736    741    742#   791    796    797#   839
                    844    845#   956    961    962#   990    995    996#  1013   1018   1019#  1053   1058
                   1059#  1088   1093   1094#  1125   1130   1131#  1160   1165   1166#  1185   1190   1191#
                   1214   1219   1220#  1241   1246   1247#  1267   1272   1273#  1298   1303   1304#  1339
                   1344   1345#  1392   1397   1398#  1424   1429   1430#  1451   1456   1457#  1494   1499
                   1500#  1531   1536   1537#  1566   1571   1572#  1591   1596   1597#  1621   1626   1627#
                   1648   1653   1654#  1674   1679   1680#  1705   1710   1711#  1746   1751   1752#  1791
                   1796   1797#  1821   1826   1827#  1850   1855   1856#  1883   1888   1889#  1925   1930
                   1931#  1966   1971   1972#  2004   2009   2010#  2054   2059   2060#  2096   2101   2102#
                   2140   2145   2146#  2210   2215   2216#  2270   2275   2276#  2304   2309   2310#  2341
                   2346   2347#  2377   2382   2383#  2410   2415   2416#  2466   2471   2472#  2523   2528
                   2529#  2560   2565   2566#  2589   2594   2595#  2617   2622   2623#  2648   2653   2654#
                   2674   2679   2680#  2710   2715   2716#  2749   2754   2755#  2804   2809   2810#  2831
                   2836   2837#  2893   2898   2999#  2958   2963   2964#  3002   3007   3008#  3036   3041
                   3042#  3067   3072   3073#  3096   3101   3102#  3127   3132   3133#  3153   3158   3159#
```

```
                    3189    3194    3195#   3228    3233    3234#   3277    3282    3283#   3305    3310    3311#   3338
                    3343    3344#   3368    3373    3374#   3435    3440    3441#   3481    3486    3487#   3521    3527
                    3528#   3560    3565    3566#   3638    3639#
$TSTM   000132       381#
$UNIT   000110       346#
$UNITM  000136       383#
$USWR   000122       353#
.     = 017362       290#    296     300     301#    303#    305#    319#    368     369#    371#    373#    385#    387#
                     389#    391#    393#    395#    397#    399#    409#    411#    422#    432#    810     1977#   1978#
                    1980#   1981#   1988#   1989#   1991#   1992#   3535#   3536#   3538#   3539#   3548#   3549#   3551#
                    3552#
.TYPE = 000004       335#    3647    3668
.$X   = 000126       368#    373
..A   = 016352      1977#   1980#   1988#   1991#   3535#   3538#   3548#   3551#
..B   = 016356      1977#   1978    1980#   1981    1988#   1989    1991#   1992    3535#   3536    3538#   3539    3548#
                    3549    3551#   3552
..C   = 000067      1977#   1980#   1988#   1991#   3535#   3538#   3548#   3551#
```

COMMEN	1#													
ENDCOM	1#													
ERROR	313#	465	489	513	538	563	589	611	625	630	634	638	642	657	666
	671	677	693	702	707	712	721	726	733	747	752	757	762	770	774
	779	787	801	806	811	816	823	829	836	851	856	865	870	875	881
	891	900	905	910	916	919	925	931	939	946	952	983	997	1026	1032
	1037	1043	1050	1060	1071	1078	1095	1112	1132	1142	1150	1173	1182	1200	1209
	1228	1238	1248	1259	1281	1295	1305	1320	1329	1346	1356	1366	1377	1399	1415
	1437	1442	1448	1467	1474	1488	1501	1518	1538	1548	1556	1579	1588	1606	1615
	1635	1645	1655	1666	1688	1702	1712	1727	1736	1753	1763	1773	1784	1806	1815
	1836	1847	1877	1896	1904	1914	1922	1932	1942	1949	1958	1982	1996	2017	2025
	2038	2047	2077	2092	2119	2134	2155	2168	2178	2188	2206	2234	2244	2264	2291
	2300	2323	2331	2365	2371	2393	2398	2404	2417	2431	2438	2455	2460	2495	2511
	2530	2543	2552	2574	2583	2605	2614	2632	2642	2655	2667	2690	2704	2717	2733
	2742	2756	2767	2777	2788	2811	2846	2851	2870	2876	2886	2913	2920	2929	2943
	2955	2965	2986	3009	3020	3028	3052	3061	3083	3093	3111	3121	3134	3146	3169
	3183	3196	3212	3221	3235	3246	3256	3267	3293	3302	3321	3332	3365	3384	3391
	3400	3420	3429	3451	3457	3466	3478	3494	3502	3507	3515	3543	3556	3575	3584
	3593	3600	3606	3616	3626	3640	3694	3703	3712	3722	3732	3742	3753	3762	3772
	3781	3790	3800												
ESCAPE	1#													
GETPRI	1#													
GETSWR	1#													
HLT	310#	464	488	512	537	562	588	610	625	630	634	638	642	657	665
	671	676	693	701	707	712	721	726	732	747	751	757	762	770	773
	779	786	801	806	811	816	823	829	835	850	856	864	870	875	881
	890	910	915	919	925	931	939	946	951	982	997	1026	1032	1037	1043
	1049	1060	1071	1078	1095	1112	1132	1142	1150	1173	1181	1200	1208	1228	1237
	1248	1259	1281	1294	1305	1320	1329	1346	1356	1366	1377	1399	1415	1437	1442
	1447	1467	1474	1487	1501	1518	1538	1548	1556	1579	1587	1606	1614	1635	1644
	1655	1666	1688	1701	1712	1727	1736	1753	1763	1773	1784	1806	1814	1836	1846
	1876	1896	1904	1914	1921	1932	1942	1949	1958	1982	1995	2017	2025	2038	2046
	2076	2091	2118	2133	2155	2168	2177	2188	2205	2234	2243	2263	2290	2299	2323
	2330	2365	2370	2393	2398	2403	2417	2431	2438	2455	2460	2495	2510	2530	2543
	2552	2574	2582	2605	2613	2632	2641	2655	2667	2690	2703	2717	2733	2742	2756
	2767	2777	2788	2811	2846	2851	2870	2876	2885	2913	2920	2929	2943	2954	2965
	2986	3009	3020	3028	3052	3060	3083	3092	3111	3120	3134	3146	3169	3182	3196
	3212	3221	3235	3246	3256	3267	3293	3301	3321	3331	3364	3384	3391	3400	3420
	3428	3451	3457	3466	3477	3494	3502	3507	3514	3543	3555	3575	3584	3593	3600
	3606	3616	3625	3640	3693	3702	3711	3721	3731	3741	3752	3761	3771	3780	3789
	3799														
HLT1	312#	899	905												
MFPS	310#	1979	1990	3537	3550										
MTPS	308#	1976	1987	3534	3547										
MULT	1#														
NEWTST	1#														
NWTEST	314#	442	471	493	519	542	569	592	618	646	682	736	791	839	956
	990	1013	1053	1088	1125	1160	1185	1214	1241	1267	1298	1339	1392	1424	1451
	1494	1531	1566	1591	1621	1648	1674	1705	1746	1791	1821	1850	1883	1925	1966
	2004	2054	2096	2140	2210	2270	2304	2341	2377	2410	2466	2523	2560	2589	2617
	2648	2674	2710	2749	2804	2831	2893	2958	3002	3036	3067	3096	3127	3153	3189
	3228	3277	3305	3338	3368	3435	3481	3521	3560						
POP	1#														
PUSH	1#														
REPORT	1#														
SEQCHK	317#	446	475	497	523	546	573	596	622	650	686	740	795	843	960

	994	1017	1057	1092	1129	1164	1189	1218	1245	1271	1302	1343	1396	1428	1455
	1498	1535	1570	1595	1625	1652	1678	1709	1750	1795	1825	1854	1887	1929	1970
	2008	2058	2100	2144	2214	2274	2308	2345	2381	2414	2470	2527	2564	2593	2621
	2652	2678	2714	2753	2808	2835	2897	2962	3006	3040	3071	3100	3131	3157	3193
	3232	3281	3309	3342	3372	3439	3485	3526	3564	3637					
SEQUEN	316#	447	476	498	524	547	574	597	623	651	687	741	796	844	961
	995	1018	1058	1093	1130	1165	1190	1219	1246	1272	1303	1344	1397	1429	1456
	1499	1[illegible]36	1571	1596	1626	1653	1679	1710	1751	1796	1826	1855	1888	1930	1971
	2009	2[illegible]9	2101	2145	2215	2275	2309	2346	2382	2415	2471	2528	2565	2594	2622
	2653	2[illegible]9	2715	2754	2809	2836	2898	2963	3007	3041	3072	3101	3132	3158	3194
	3233	3282	3310	3343	3373	3440	3486	3527	3565	3638					
SETPRI	1#														
SETUP	1#														
SKIP	1#														
SLASH	1#														
STARS	1#	272#	292	298	316#	338	365	367	374	442	444	471	473	493	495
	519	521	542	544	569	571	592	594	618	620	646	648	682	684	736
	738	791	793	839	841	956	958	990	992	1013	1015	1053	1055	1088	1090
	1125	1127	1160	1162	1185	1187	1214	1216	1241	1243	1267	1269	1298	1300	1339
	1341	1392	1394	1424	1426	1451	1453	1494	1496	1531	1533	1566	1568	1591	1593
	1621	1623	1648	1650	1674	1676	1705	1707	1746	1748	1791	1793	1821	1823	1850
	1852	1883	1885	1925	1927	1966	1968	2004	2006	2054	2056	2096	2098	2140	2142
	2210	2212	2270	2272	2304	2306	2341	2343	2377	2379	2410	2412	2466	2468	2523
	2525	2560	2562	2589	2591	2617	2619	2648	2650	2674	2676	2710	2712	2749	2751
	2804	2806	2831	2833	2893	2895	2958	2960	3002	3004	3036	3038	3067	3069	3096
	3098	3127	3129	3153	3155	3189	3191	3228	3230	3277	3279	3305	3307	3338	3340
	3368	3370	3435	3437	3481	3483	3521	3523	3560	3562	3658				
SWRSU	1#														
TYPBIN	1#														
TYPDEC	1#														
TYPNAM	1#														
TYPNUM	1#														
TYPOCS	1#														
TYPOCT	1#														
TYPTXT	1#														
$C17	307#	660	696	859	894										
$NWTST	315#	442	471	493	519	542	569	592	618	646	682	736	791	839	956
	990	1013	1053	1088	1125	1160	1185	1214	1241	1267	1298	1339	1392	1424	1451
	1494	1531	1566	1591	1621	1648	1674	1705	1746	1791	1821	1850	1883	1925	1966
	2004	2054	2096	2140	2210	2270	2304	2341	2377	2410	2466	2523	2560	2589	2617
	2648	2674	2710	2749	2804	2831	2893	2958	3002	3036	3067	3096	3127	3153	3189
	3228	3277	3305	3338	3368	3435	3481	3521	3560						
$$ESCA	1#														
$$NEWT	1#														
$$SKIP	1#														
.EQUAT	1#														
.HEADE	1#	272#													
.KT11	1#														
.SETUP	1#														
.SWRHI	1#														
.$ACT1	1#	272#	296												
.$APTB	1#	272#	336												
.$APTH	1#	272#	363												
.$APTY	1#														
.$ASTA	1#														
.$CATC	1#														

```
.$CMTA      1#
.$DB2D      1#
.$DB2O      1#
.$DIV       1#
.$EOP       1#
.$ERRO      1#
.$ERRT      1#
.$MULT      1#
.$POWE      1#
.$RAND      1#
.$RDDE      1#
.$RDOC      1#
.$READ      1#
.$R2AZ      1#
.$SAVE      1#
.$SB2D      1#
.$SB2O      1#
.$SCOP      1#
.$SIZE      1#
.$SUPR      1#
.$TRAP      1#
.$TYPB      1#
.$TYPD      1#
.$TYPE      1#
.$TYPO      1#
.$40CA      1#
.1170       1#

. ABS.  017362     000

ERRORS DETECTED:  0

CVKAAC.BIN,CVKAAC.LST/CRF/SOL/NL:TOC=CVKAAC.SML,CVKAAC.P11
RUN-TIME: 13 18 1 SECONDS
RUN-TIME RATIO: 103/33=3.0
CORE USED:  32K  (63 PAGES)
```