LSI

MACRO INSTRUCTION EXERCISER

MD-11-DVKAA-A

EP-DVKAA-A-DL-A

OCT 1976

FICHE 1 OF 1

digital

Made in U.S.A.

```
 4888
```

E01


```
 4890
```

```
                                        .ABS
                                        ;; LSI-11 MACRO INSTRUCTION EXERCISER
                                        .NLIST  MC,MD,CND
                                        .LIST   ME
                                        .MCALL  .HEADER,.$APTHDR,.$APTBLS,.$ACT11,STARS
                                .TITLE  DVKAAA
                                ;*COPYRIGHT (C) AUGUST 1975
                                ;*DIGITAL EQUIPMENT CORP.
                                ;*MAYNARD, MASS. 01754
                                ;*
                                ;*PROGRAM BY PERVEZ ZAKI
                                ;*
                                ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
                                ;*PACKAGE (MAINDEC-11-DZQAC-B),JULY 11,1975.
                                ;*
  (1)           000001          $TN=1
  (1)           160000          $SWR=160000     ;;HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYPOUT
```

```
4906            000000                          .=0
4907
4908                                    ;*****************************************************************
4909
4910                                    ;       TRAP CATCHERS OF .+2 AND HALT IN LOCATIONS 0 THRU 776 [IT IS NLISTED]
4911
4918                                    ;*****************************************************************
 (1)
 (1)                                    .SBTTL  ACT11 HOOKS
 (1)                                    ;HOOKS REQUIRED BY ACT11
 (1)            001000                          $SVPC=.                 ;SAVE PC
 (1)            000046                          .=46
 (1)  000046    016750                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
 (1)            000052                          .=52
 (1)  000052    000000                          .WORD   0               ;;2)SET LOC.52 TO ZERO
 (1)            001000                          .=$SVPC                 ;; RESTORE PC
4919
4932
4946
4947
4967
4968
4975
4981
4985
4997
5004
5011
5012
5013            000400                  .=400
5014            000000                  R0      =%0
5015            000001                  R1      =%1
5016            000002                  R2      =%2
5017            000003                  R3      =%3
5018            000004                  R4      =%4
5019            000005                  R5      =%5
5020            000006                  R6      =%6
5021            000006                  SP      =%6
5022            000007                  PC      =%7
5023            000254                  CLNZ    =254
5024            000001                  ERRNM   =1
5025            000260                  NOP1    =260
5026            000263                  SEVC    =263
5027            000273                  SENVC   =273
5028            000000                  $TN     =0
5029            000004                  .TYPE   =IOT
```

```
                                ;;*******************************************************************
(1)
(1)                             .SBTTL  APT MAILBOX-ETABLE
(1)
(1)
(1)
(1)
(1)                                     .EVEN
(1)     000400                  $MAIL:                          ;;APT MAILBOX
(1)     000400  000000          $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
(1)     000402  000000          $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
(1)     000404  000000          $TESTN: .WORD   ATESTN          ;;TEST NUMBER
(1)     000406  000000          $PASS:  .WORD   APASS           ;;PASS COUNT
(1)     000410  000000          $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
(1)     000412  000000          $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
(1)     000414  000000          $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
(1)     000416  000000          $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
(1)     000420                  $ETABLE:                        ;;APT ENVIRONMENT TABLE
(1)     000420     000          $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
(1)     000421     000          $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
(1)     000422  000000          $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
(1)     000424  000000          $USWR:  .WORD   AUSWR           ;;USER SWITCHES
(1)     000426  000000          $CPUOP: .WORD   ACPUOP          ;;CPU TYPE,OPTIONS
(1)                             ;*                              BITS 15-11=CPU TYPE
(1)                             ;*                                      11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
(1)                             ;*                                      11/70=06,PDQ=07,Q=10
(1)                             ;*                              BIT 10=REAL TIME CLOCK
(1)                             ;*                              BIT  9=FLOATING POINT PROCESSOR
(1)                             ;*                              BIT  8=MEMORY MANAGEMENT
(1)     000430                  $ETEND:
(1)                                     .MEXIT
                                ;;*******************************************************************
(1)
(1)                             .SBTTL  APT PARAMETER BLOCK
(1)                             ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
(2)                             ;;*******************************************************************
(1)             000430                  .$X=.   ;;SAVE CURRENT LOCATION
(1)             000024                  .=24    ;;SET POWER FAIL TO POINT TO START OF PROGRAM
(1)     000024  000200                  200     ;;FOR APT START UP
(1)             000044                  .=44    ;;POINT TO APT INDIRECT ADDRESS PNTR.
(1)     000044  000430                  $APTHDR ;;POINT TO APT HEADER BLOCK
(1)             000430                  .=.$X   ;;RESET LOCATION COUNTER
(2)                             ;;*******************************************************************
(1)                             ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
(1)                             ;INTERFACE SPEC.
(1)
(1)     000430                  $APTHD:
(1)     000430  000000          $HIBTS: .WORD   0               ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
(1)     000432  000400          $MBADR: .WORD   $MAIL           ;;ADDRESS OF APT MAILBOX (BITS 0-15)
(1)     000434  000001          $TSTM:  .WORD   1               ;;RUN TIM OF LONGEST TEST
(1)     000436  000001          $PASTM: .WORD   1               ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
(1)     000440  000000          $UNITM: .WORD                   ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
(1)     000442  000014                  .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```

```
5034                    000430                          .=$APTHD
5035    000430                                          ADR:
5036                    000432                          .=ADR+2
5037    000432                                          ADR1:
5038                    000434                          .=ADR1+2
5039    000434                                          ADR2:
5040                    000436                          .=ADR2+2
5041    000436                                          DUMMY:
5042                    000440                          .=DUMMY+2
5043    000440                                          TEMP:
5044                    000442                          .=TEMP+2
5045    000442                                          TEMP1:
5046                    000444                          .=TEMP1+2
5047    000444                                          TEMP2:
5048                    000446                          .=TEMP2+2
5049    000446  177564                                  TPS:    177564          ; OUTPUT TTY STATUS REGISTER
5050    000450  177566                                  TPB:    177566          ; OUTPUT BUFFER
5051    000452  006402                          MARK2:  MARK    2
5052    000454  005015  042440  042116  ENDPAS: .ASCIZ  <15><12>" END PASS"
        000462  050040  051501  000123
5053                                            .EVEN
5054    000470  005015  047520  042527  POWER:  .ASCIZ  <15><12>/POWER/
        000476  000122
5055                                            .EVEN
5056    000500  000023                          .BLKW   19.
5057
5058                    000020                  .=20
5059    000020  017020                          TYPE
5060    000022  000000                          0
5061
5062
5063
```

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


               000200                           .=200
      000200   012737   016770   000024         MOV     #PWRDN,@#24     ; SERVICE POWER DOWN ROUTINE ON ANY FUTURE POWER DOWN
      000206   012700   000420                  MOV     #$ETABLE,R0
      000212   005040                     2$:   CLR     -(R0)           ;START CLEANING THE STACK
      000214   020027   000400                  CMP     R0,#$MAIL       ;FOR INITIALIZATION
      000220   101374                           BHI     2$
      000222   000167   000302                  JMP     START
                                          ;     --------------------------


               000530                           .=530

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

```
5093                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 0       CHECK BRANCH INSTRUCTIONS WITH ZERO CONDITION CODES
 (3)                                    ;;*******************************************************************************
5094
5095  000546                            NOBIT:
 (2)  000546  021527  000000                    CMP     (R5),#0
5096  000552  001017                            BNE     CC0             ; IF IN WRONG SEQUENCE GO TO HALT AT END OF THE TEST
5097  000554  005215                    1$:     INC     (R5)
5098  000556  000257                            CCC                     ; ZERO CONDITION CODES, NZVC=0000
5099  000560  103414                            BCS     CC0
5100  000562  102413                            BVS     CC0
5101  000564  001412                            BEQ     CC0
5102  000566  100411                            BMI     CC0
5103  000570  000260                            NOP1                    ; CHECK NOP1 INSTRUCTION I.E. OP-CODE 260
5104  000572  103407                            BCS     CC0
5105  000574  102406                            BVS     CC0
5106  000576  001405                            BEQ     CC0
5107  000600  100404                            BMI     CC0
5108  000602  002403                            BLT     CC0
5109  000604  003402                            BLE     CC0
5110  000606  101401                            BLOS    CC0
5111  000610  101004                            BHI     ENDCC0
5112  000612                            CC0:
 (2)  000612  012745  000001                    MOV     #1,-(R5)
 (2)  000616  005245                            INC     -(R5)
 (2)  000620  000000                            HALT                    ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
5113  000622  102000                    ENDCC0: BVC     NBIT
5114
5115
5116                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 1       CHECK BRANCH INSRUCTIONS WITH N BIT SET
 (3)                                    ;;*******************************************************************************
5117
5118  000624                            NBIT:
 (2)  000624  021527  000001                    CMP     (R5),#1
5119  000630  001012                            BNE     CC1             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THETEST
5120  000632  005215                    1$:     INC     (R5)
5121  000634  000270                            SEN                     ; NBIT IS SET, NZVC=1000
5122  000636  100007                            BPL     CC1
5123  000640  001406                            BEQ     CC1
5124  000642  002005                            BGE     CC1
5125  000644  003004                            BGT     CC1
5126  000646  103403                            BCS     CC1
5127  000650  101402                            BLOS    CC1
5128  000652  103401                            BLO     CC1
5129  000654  003404                            BLE     ENDCC1
5130  000656                            CC1:
 (2)  000656  012745  000002                    MOV     #2,-(R5)
 (2)  000662  005245                            INC     -(R5)
 (2)  000664  000000                            HALT                    ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
5131  000666  001000                    ENDCC1: BNE     VBIT
```

```
5132                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 2       CHECK BRANCH INSRUCTIONS WITH N&V BITS SET
 (3)                                    ;;*******************************************************************************
5133
5134  000670                            VBIT:
 (2)  000670  021527  000002                    CMP     (R5),#2
5135  000674  001014                            BNE     CC2             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
5136  000676  005215                    1$:     INC     (R5)
5137  000700  000270                            SEN
5138  000702  000262                            SEV                     ; V AND N BIT SET, NZVC = 1010
5139  000704  102010                            BVC     CC2
5140  000706  001407                            BEQ     CC2
5141  000710  100006                            BPL     CC2
5142  000712  103405                            BCS     CC2
5143  000714  002404                            BLT     CC2
5144  000716  003403                            BLE     CC2
5145  000720  101402                            BLOS    CC2
5146  000722  103401                            BLO     CC2
5147  000724  003004                            BGT     ENDCC2
5148  000726                            CC2:
 (2)  000726  012745  000003                    MOV     #3,-(R5)
 (2)  000732  005245                            INC     -(R5)
 (2)  000734  000000                            HALT                    ; ONE OF THE ABOVE BRANCHES FAILED OR WRONG SEQUENCE
5149  000736  002000                    ENDCC2: BGE     CBIT
5150
5151
5152                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 3       CHECK BRANCH INSRUCTIONS WITH N,V&C BITS SET
 (3)                                    ;;*******************************************************************************
5153
5154  000740                            CBIT:
 (2)  000740  021527  000003                    CMP     (R5),#3
5155  000744  001013                            BNE     CC3             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
5156  000746  005215                    1$:     INC     (R5)
5157  000750  000270                            SEN
5158  000752  000262                            SEV
5159  000754  000261                            SEC                     ; C, V, AND N BITS ARE SET, NZVC=1011
5160  000756  001406                            BEQ     CC3
5161  000760  100005                            BPL     CC3
5162  000762  102004                            BVC     CC3
5163  000764  002403                            BLT     CC3
5164  000766  003402                            BLE     CC3
5165  000770  101001                            BHI     CC3
5166  000772  002004                            BGE     ZBIT
5167  000774                            CC3:
 (2)  000774  012745  000004                    MOV     #4,-(R5)
 (2)  001000  005245                            INC     -(R5)
 (2)  001002  000000                            HALT                    ; ONE OF THE ABOVE BRANCHES FAILED
5168                                                                    ; OR WRONG SEQUENCE
```

```
5169                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 4       CHECK BRANCH INSRUCTIONS WITH N,Z,V&C BITS SET
 (3)                                    ;;*******************************************************************************
5170
5171  001004                            ZBIT:
 (2)  001004  021527  000004                    CMP     (R5),#4
5172  001010  001015                            BNE     CC4             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
5173  001012  005215                            INC     (R5)
5174  001014  000270                            SEN
5175  001016  000262                            SEV
5176  001020  000261                            SEC
5177  001022  000264                            SEZ                     ; ALL BITS SET, NZVC=1111
5178  001024  001007                            BNE     CC4
5179  001026  100006                            BPL     CC4
5180  001030  102005                            BVC     CC4
5181  001032  103004                            BCC     CC4
5182  001034  002403                            BLT     CC4
5183  001036  003002                            BGT     CC4
5184  001040  101001                            BHI     CC4
5185  001042  001404                            BEQ     YESCC
5186  001044                            CC4:
 (2)  001044  012745  000005                    MOV     #5,-(R5)
 (2)  001050  005245                            INC     -(R5)
 (2)  001052  000000                            HALT                    ; ONE OF THE ABOVE BRANCHES FAILED
5187                                                                    ; OR WRONG SEQUENCE
5188
5189
5190                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 5       CHECK BRANCH INSRUCTIONS WITH ALL THE CONDITION CODES SET
 (3)                                    ;;*******************************************************************************
5191
5192  001054                            YESCC:
 (2)  001054  021527  000005                    CMP     (R5),#5
5193  001060  001014                            BNE     CC6             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
5194  001062  005215                            INC     (R5)
5195  001064  000277                            SCC                     ; NZVC=1111
5196  001066  100011                            BPL     CC6
5197  001070  001010                            BNE     CC6
5198  001072  102007                            BVC     CC6
5199  001074  103006                            BCC     CC6
5200  001076  000240                            NOP                     ; CHECK NOP INSTRUCTION
5201  001100  100004                            BPL     CC6
5202  001102  001003                            BNE     CC6
5203  001104  102002                            BVC     CC6
5204  001106  103001                            BCC     CC6
5205  001110  101404                            BLOS    NOTCC
5206  001112                            CC6:
 (2)  001112  012745  000006                    MOV     #6,-(R5)
 (2)  001116  005245                            INC     -(R5)
 (2)  001120  000000                            HALT                    ; SCC OR A BRANCH FAILED, OR WRONG SEQUENCE
```

```
5207                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 6       CLEAR THE CONDITION CODES
 (3)                                    ;;*******************************************************************************
5208
5209  001122                            NOTCC:
 (2)  001122  021527  000006                    CMP     (R5),#6
5210  001126  001013                            BNE     CC5             ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
5211  001130  005215                            INC     (R5)
5212  001132  000277                            SCC                     ; NZVC=1111
5213  001134  000241                            CLC                     ; NZVC=1110
5214  001136  103407                            BCS     CC5
5215  001140  000242                            CLV                     ; NZVC=1100
5216  001142  102405                            BVS     CC5
5217  001144  000244                            CLZ                     ; NZVC=1000
5218  001146  001403                            BEQ     CC5
5219  001150  000250                            CLN                     ; NZVC=0000
5220  001152  100401                            BMI     CC5
5221  001154  101004                            BHI     ENDCC5
5222  001156                            CC5:
 (2)  001156  012745  000007                    MOV     #7,-(R5)
 (2)  001162  005245                            INC     -(R5)
 (2)  001164  000000                            HALT                    ; ONE OF THE ABOVE CLEARS FAILED OR WRONG SEQUENCE
5223  001166  100000                    ENDCC5: BPL     BRANCH
5224
5225
5226
5227                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 7       CHECK FORWARD AND BACKWARD BRANCHES.
 (3)                                    ;;*******************************************************************************
5228
5229  001170                            BRANCH:
 (2)  001170  021527  000007                    CMP     (R5),#7
5230  001174  001404                            BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
5231  001176  012745  000010                    MOV     #10,-(R5)
 (2)  001202  005245                            INC     -(R5)
 (2)  001204  000000                            HALT
5232  001206  005215                    1$:     INC     (R5)
5233  001210  000416                            BR      4$              ; CHECK BRANCH FORWARD AND BACKWARD
5234  001212  012745  000011                    MOV     #11,-(R5)
 (2)  001216  005245                            INC     -(R5)
 (2)  001220  000000                            HALT                    ; FORWARD BRANCH FAILED
5235  001222  000404                    2$:     BR      3$
5236  001224  012745  000012                    MOV     #12,-(R5)
 (2)  001230  005245                            INC     -(R5)
 (2)  001232  000000                            HALT                    ; FORWARD BRANCH FAILED
5237  001234  000411                    3$:     BR      5$
5238  001236  012745  000013                    MOV     #13,-(R5)
 (2)  001242  005245                            INC     -(R5)
 (2)  001244  000000                            HALT                    ; FORWARD BRANCH FAILED
5239  001246  000765                    4$:     BR      2$
5240  001250  012745  000014                    MOV     #14,-(R5)
 (2)  001254  005245                            INC     -(R5)
 (2)  001256  000000                            HALT                    ; BACKWARD BRANCH FAILED
5241  001260  000400                    5$:     BR      JMP1
```

```
5242                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 10      CHECK JMP INSTRUCTIONS FOR MODE 1
 (3)                                    ;;*******************************************************************************
5243
5244  001262                           JMP1:
 (2)  001262  021527  000010                    CMP     (R5),#10
5245  001266  001033                           BNE     ENDJP1          ; IF IN WRONG SEQUENCE GO TO HALT AT THE END OF THE TEST
5246  001270  005215                    1$:    INC     (R5)
5247  001272  012700  001312                   MOV     #2$,R0          ; TEST JUMP INSTRUCTION MODE 1
5248  001276  000277                           SCC
5249  001300  000110                           JMP     (R0)
5250  001302  012745  000015                   MOV     #15,-(R5)
 (2)  001306  005245                           INC     -(R5)
 (2)  001310  000000                           HALT                    ; JUMP INSTRUCTION FAILED
5251  001312                           2$:
 (1)  001312  100003                           BPL     3$
 (1)  001314  001002                           BNE     3$
 (1)  001316  102001                           BVC     3$
 (1)  001320  103404                           BCS     4$
 (2)  001322                            3$:
 (3)  001322  012745  000016                   MOV     #16,-(R5)
 (3)  001326  005245                           INC     -(R5)
 (3)  001330  000000                           HALT                    ; WRONG CC
5252  001332  020027  001312           4$:    CMP     R0,#2$
5253  001336  001404                           BEQ     5$              ; CONTINUE IF R0 IS OK
5254  001340  012745  000017                   MOV     #17,-(R5)
 (2)  001344  005245                           INC     -(R5)
 (2)  001346  000000                           HALT
5255  001350  012700  001366           5$:    MOV     #JMP2,R0        ; TEST JUMP INSTRUCTION MODE 1
5256  001354  000110                           JMP     (R0)
5257  001356                           ENDJP1:
 (2)  001356  012745  000020                   MOV     #20,-(R5)
 (2)  001362  005245                           INC     -(R5)
 (2)  001364  000000                           HALT                    ; JUMP INSTRUCTION FAILED OR WRONG SEQUENCE
5258
5259
5260                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 11      CHECK JMP INSTRUCTIONS FOR MODES 2 AND 3
 (3)                                    ;;*******************************************************************************
5261
5262  001366                           JMP2:
 (2)  001366  021527  000011                    CMP     (R5),#11
5263  001372  001073                           BNE     ENDJP3          ; IF IN WRONG SEQUENCE GO TO HALT AT THE END OF TEST
5264  001374  005215                           INC     (R5)
5265  001376  012700  001416                   MOV     #3$,R0          ; TEST JUMP INSTRUCTION MODE 2
5266  001402  000277                           SCC
5267  001404  000120                           JMP     (R0)+
5268  001406  012745  000021                   MOV     #21,-(R5)
 (2)  001412  005245                           INC     -(R5)
 (2)  001414  000000                           HALT                    ; JUMP INSTRUCTION FAILED
5269  001416                           3$:
 (1)  001416  100003                           BPL     4$
 (1)  001420  001002                           BNE     4$
 (1)  001422  102001                           BVC     4$
 (1)  001424  103404                           BCS     5$
 (2)  001426                            4$:
```

```
   (3)  001426  012745  000022                    MOV     #22,-(R5)
   (3)  001432  005245                            INC     -(R5)
   (3)  001434  000000                            HALT
                                                  ; WRONG CC
5270    001436  020027  001420          5$:       CMP     R0,#3$+2        ; IS THERE AUTO INC.?
5271    001442  001404                            BEQ     6$
5272    001444  012745  000023                    MOV     #23,-(R5)
   (2)  001450  005245                            INC     -(R5)
   (2)  001452  000000                            HALT
                                                  ; MODE 2 FAILED FOR JMP INSTRUCTION
5273    001454  012700  001472          6$:       MOV     #JMP3,R0        ; TEST JUMP INSTRUCTION MODE 2
5274    001460  000120                            JMP     (R0)+
5275    001462  012745  000024                    MOV     #24,-(R5)
   (2)  001466  005245                            INC     -(R5)
   (2)  001470  000000                            HALT
                                                  ; JUMP INSTRUCTION FAILED
5276
5277    001472  012767  001526  176740  JMP3:     MOV     #3$,TEMP        ; TEST JUMP INSTRUCTION MODE 3
5278    001500  012767  001546  176734            MOV     #4$,TEMP+2      ;
5279    001506  012700  000440                    MOV     #TEMP,R0
5280    001512  000277                            SCC
5281    001514  000130                            JMP     @(R0)+          ;
5282    001516  012745  000025                    MOV     #25,-(R5)
   (2)  001522  005245                            INC     -(R5)
   (2)  001524  000000                            HALT
                                                  ; JUMP INSTRUCTION FAILED
5283    001526  027067  000000  000012  3$:       CMP     @(R0),4$        ; IS THERE AUTO INC.?
5284    001534  001404                            BEQ     4$
5285    001536  012745  000026                    MOV     #26,-(R5)
   (2)  001542  005245                            INC     -(R5)
   (2)  001544  000000                            HALT
                                                  ; JMP INSTRUCTION FAILED IN MODE 2
5286    001546  012767  001572  176664  4$:       MOV     #JMP4,TEMP      ; TEST JUMP INSTRUCTION MODE 3
5287    001554  012700  000440                    MOV     #TEMP,R0
5288    001560  000130                            JMP     @(R0)+
5289    001562                           ENDJP3:
   (2)  001562  012745  000027                    MOV     #27,-(R5)
   (2)  001566  005245                            INC     -(R5)
   (2)  001570  000000                            HALT
                                                  ; JUMP ERROR OR WRONG SEQUENCE
```

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

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

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


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

```
  (2)                                   ;*TEST: 13      TEST JMP INSTRUCTION FOR MODE 6 AND 7
  (3)                                   ;;*************************************************************************
 5324
 5325   002002                         JMP6:
  (2)   002002  021527  000013                 CMP     (R5),#13
 5326   002006  001071                        BNE     ENDJP7          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
 5327   002010  005215                        INC     (R5)
 5328   002012  012703  002040                MOV     #1$+6,R3
 5329   002016  000163  177772                JMP     -6(R3)
 5330   002022  012745  000040                MOV     #40,-(R5)
  (2)   002026  005245                        INC     -(R5)
  (2)   002030  000000                        HALT                    ; JUMP INSTRUCTION FAILED
 5331   002032  020327  002040        1$:     CMP     R3,#1$+6        ; CHECK R3
 5332   002036  001404                        BEQ     2$
 5333   002040  012745  000041                MOV     #41,-(R5)
  (2)   002044  005245                        INC     -(R5)
  (2)   002046  000000                        HALT                    ; WRONG VALUE IN REGISTER AFTER JUMP MODE 6
 5334                                                                  ; OR JUMP INSTRUCTION FAILED IN MODE 6
 5335   002050  000167  000010        2$:     JMP     3$-.-4(PC)      ; TEST JUMP INSTRUCTION MODE 6
 5336   002054  012745  000042                MOV     #42,-(R5)
  (2)   002060  005245                        INC     -(R5)
  (2)   002062  000000                        HALT                    ; JUMP INSTRUCTION FAILED
 5337   002064  012703  002104        3$:     MOV     #JMP7,R3        ; JUMP SHOULD LAND HERE
 5338   002070  000163  000000                JMP     0(R3)
 5339   002074  012745  000043                MOV     #43,-(R5)
  (2)   002100  005245                        INC     -(R5)
  (2)   002102  000000                        HALT                    ; JUMP INSTRUCTION FAILED
 5340
 5341   002104  012703  000440        JMP7:   MOV     #TEMP,R3
 5342   002110  012713  002130                MOV     #1$,(R3)
 5343   002114  000173  000000                JMP     @(R3)
 5344   002120  012745  000044                MOV     #44,-(R5)
  (2)   002124  005245                        INC     -(R5)
  (2)   002126  000000                        HALT                    ; JUMP INSTRUCTION FAILED
 5345   002130  012713  002154        1$:     MOV     #3$,(R3)        ; TEST JUMP INSTRUCTION MODE 7
 5346   002134  012700  000434                MOV     #TEMP-4,R0
 5347   002140  000170  000004                JMP     @4(R0)
 5348   002144  012745  000045                MOV     #45,-(R5)
  (2)   002150  005245                        INC     -(R5)
  (2)   002152  000000                        HALT                    ; JUMP INSTRUCTION FAILED
 5349   002154  012767  002202  176256  3$:     MOV     #JSRTST,TEMP    ; CONTINUE
 5350   002162  012700  000440                MOV     #TEMP,R0
 5351   002166  000170  000000                JMP     @0(R0)
 5352   002172                         ENDJP7:
  (2)   002172  012745  000046                MOV     #46,-(R5)
  (2)   002176  005245                        INC     -(R5)
  (2)   002200  000000                        HALT                    ; JUMP ERROR OR SEQUENCE ERROR
```

```
                                            ;;*******************************************************************
 (2)                                        ;*TEST: 14      CHECK JSR AND MARK INSTRUCTIONS
 (3)                                        ;;*******************************************************************

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

```
   (1)  002400  100003                        BPL     3$
   (1)  002402  001002                        BNE     3$
   (1)  002404  102001                        BVC     3$
   (1)  002406  103404                        BCS     4$
   (2)  002410                          3$:
   (3)  002410  012743  000056                MOV     #56,-(R3)
   (3)  002414  005243                        INC     -(R3)
   (3)  002416  000000                        HALT                    ; WRONG CC
5384    002420  022705  000404          4$:   CMP     #$TESTN,R5
5385    002424  001404                        BEQ     5$
5386    002426  012743  000057                MOV     #57,-(R3)
   (2)  002432  005243                        INC     -(R3)
   (2)  002434  000000                        HALT                    ; MARK INSRUCTION FAILED
5387    002436  022706  000530          5$:   CMP     #START,SP
5388    002442  001404                        BEQ     6$
5389    002444  012745  000060                MOV     #60,-(R5)
   (2)  002450  005245                        INC     -(R5)
   (2)  002452  000000                        HALT                    ; MARK INSTRUCTION FAILED
5390    002454  012701  002562          6$:   MOV     #12$,R1         ; PLACE THE ADDRESS OF 12$ IN R1
5391    002460  004011                        JSR     R0,(R1)         ; GO TO TAG 12$
5392    002462                          7$:
   (2)  002462  012745  000061                MOV     #61,-(R5)
   (2)  002466  005245                        INC     -(R5)
   (2)  002470  000000                        HALT                    ; JSR INSTRUCTION FAILED
5393    002472  012745  000062                MOV     #62,-(R5)
   (2)  002476  005245                        INC     -(R5)
   (2)  002500  000000                        HALT                    ; RTS BROUGHT THE PROGRAM BACK IN WRONG
5394                                                                   ; PLACE
5395    002502  022706  000530          8$:   CMP     #START,SP
5396    002506  001443                        BEQ     REGS
5397    002510  012745  000063                MOV     #63,-(R5)
   (2)  002514  005245                        INC     -(R5)
   (2)  002516  000000                        HALT                    ; STACK POINTER WAS NOT RESET
5398
5399    002520  020427  002370          10$:  CMP     R4,#1$          ; IS THE RETURN ADDRESS =1$ ?
5400    002524  001404                        BEQ     11$
5401    002526  012745  000064                MOV     #64,-(R5)
   (2)  002532  005245                        INC     -(R5)
   (2)  002534  000000                        HALT                    ; WRONG RETURN ADDRESS IN UNKAGE REGISTER R4
5402    002536  010605                  11$:  MOV     SP,R5
5403    002540  005725                        TST     (R5)+           ; SET UP ADDRESS IN R5 AT MARK 2 INSTRUCTION
5404    002542  012716  002400                MOV     #2$,(SP)        ; SET RETURN ADDRESS =2$
5405    002546  000277                        SCC
5406    002550  000205                        RTS     R5              ; RETURN USING R5 AND IN-TURN USING MARK INSTRUCTION
5407    002552  012745  000065                MOV     #65,-(R5)
   (2)  002556  005245                        INC     -(R5)
   (2)  002560  000000                        HALT                    ; RTS INSTRUCTION FAILED
5408
5409    002562  020027  002462          12$:  CMP     R0,#7$          ; DOES R0 CONTAIN THE RETURN ADDRESS?
5410
5411    002566  001404                        BEQ     13$
5412    002570  012745  000066                MOV     #66,-(R5)
   (2)  002574  005245                        INC     -(R5)
   (2)  002576  000000                        HALT                    ; WRONG RETURN ADDRESS IN LINKAGE REGISTER R0
5413    002600  012700  002502          13$:  MOV     #8$,R0          ; SET RETURN ADDRESS AT 8$
5414    002604  000200                        RTS     R0
```

```
 5415  002606                          ENDJSR:
  (2)  002606  012745  000067                MOV     #67,-(R5)
  (2)  002612  005245                        INC     -(R5)
  (2)  002614  000000                        HALT                    ; RTS INSTRUCTION FAILED OR SEQUENCE ERROR
 5416
 5417                                 ;;*******************************************************************************
  (2)                                 ;*TEST: 15      CHECK REGISTER SELECTION
  (3)                                 ;;*******************************************************************************
 5418
 5419  002616                          REGS:
  (2)  002616  021527  000015                MOV     (R5),#15
 5420  002622  001034                        BNE     EREGS           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
 5421  002624  005215                        INC     (R5)
 5422  002626  010667  175606                MOV     R6,TEMP         ; SAVE THE STACK POINTER
 5423  002632  012700  000001                MOV     #1,R0           ; LOAD THE REGISTERS
 5424  002636  012701  000004                MOV     #4,R1           ;
 5425  002642  012702  000020                MOV     #20,R2          ;
 5426  002646  012703  000100                MOV     #100,R3         ;
 5427  002652  012704  000400                MOV     #400,R4         ;
 5428  002656  005006                        CLR     R6              ;
 5429  002660  060006                        ADD     R0,R6           ; ADD UP THE REGISTERS
 5430  002662  060106                        ADD     R1,R6           ;
 5431  002664  060206                        ADD     R2,R6           ;
 5432  002666  060306                        ADD     R3,R6           ;
 5433  002670  060406                        ADD     R4,R6           ;
 5434  002672  060506                        ADD     R5,R6           ;
 5435  002674  022706  001131                CMP     #$TESTN+525,R6  ; CHECK IT
 5436  002700  001003                        BNE     1$              ; FAILED
 5437  002702  016706  175532                MOV     TEMP,R6         ; RESTORE STACK POINTER
 5438  002706  000406                        BR      TSTB0           ; CONTINUE
 5439  002710  016706  175524        1$:     MOV     TEMP,R6         ; RESTORE STACK POINTER
 5440  002714                          EREGS:
  (2)  002714  012745  000070                MOV     #70,-(R5)
  (2)  002720  005245                        INC     -(R5)
  (2)  002722  000000                        HALT                    ; REGISTER SELECTION FAILURE OR SEQUENCE ERROR
```

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

                                ;;*************************************************************************
                                ;*TEST: 16      NEW INSTRUCTIONS USED IN THIS SECTION ARE TSTB, CLRB, MOVB
                                ;;*************************************************************************
002724                          TSTB0:
002724  021527  000016                  CMP     (R5),#16
002730  001404                          BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
002732  012745  000071                  MOV     #71,-(R5)
002736  005245                          INC     -(R5)
002740  000000                          HALT                            ; PROGRAM IS IN WRONG SEQUENCE
002742  005215                  2$:     INC     (R5)
002744  000277                          SCC
002746  105000                          CLRB    R0                      ; CLEAR THE REGISTER
002750  004737  017164                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
002754  105700                          TSTB    R0                      ; CHECK IT
002756  004737  017164                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
002762  112701  000377                  MOVB    #377,R1                 ; LOAD THE REGISTER
002766  004737  017252                  JSR     PC,@#$CC10              ; CHECK FOR CC = 10
002772  105701                          TSTB    R1                      ; CHECK IT
002774  004737  017252                  JSR     PC,@#$CC10              ; CHECK FOR CC = 10


                                ;;*************************************************************************
                                ;*TEST: 17      NEW INSTRUCTIONS USED IN THIS SECTION ARE CMPB, BISB
                                ;;*************************************************************************
003000                          CMPB0:
003000  021527  000017                  CMP     (R5),#17
003004  001027                          BNE     ECMPB0                  ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
003006  005215                  1$:     INC     (R5)
003010  000277                          SCC
003012  152702  000377                  BISB    #377,R2                 ; LOAD REGISTER
003016  004737  017272                  JSR     PC,@#$CC11              ; CHECK FOR CC = 11
003022  122702  000377                  CMPB    #377,R2                 ; CHECK COMPARE
003026  001404                          BEQ     2$                      ; CONTINUE IF OK
003030  012745  000072                  MOV     #72,-(R5)
003034  005245                          INC     -(R5)
003036  000000                          HALT                            ; BISB OR CMPB INSTRUCTION FAILED
003040  112700  000077          2$:     MOVB    #77,R0
003044  120002                          CMPB    R0,R2                   ; CHECK IT AGAIN
003046  100004                          BPL     3$                      ; CONTINUE IF OK
003050  012745  000073                  MOV     #73,-(R5)
003054  005245                          INC     -(R5)
003056  000000                          HALT                            ; CMPB INSTRUCTION FAILED [WRONG CC]
003060  120200                  3$:     CMPB    R2,R0                   ; ONCE MORE
003062  100404                          BMI     BICB0                   ; CONTINUE IF OK
003064                          ECMPB0:
003064  012745  000074                  MOV     #74,-(R5)
003070  005245                          INC     -(R5)
003072  000000                          HALT                            ; WRONG CC OR WRONG SEQUENCE
```

```
5485                                    ;;*************************************************************************
 (2)                                    ;*TEST: 20      NEW INSTRUCTIONS USED IN THIS SECTION ARE BICB, BITB
 (3)                                    ;;*************************************************************************
5486
5487  003074                            BICB0:
 (2)  003074  021527  000020                    CMP     (R5),#20
5488  003100  001404                           BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
5489  003102  012745  000075                   MOV     #75,-(R5)
 (2)  003106  005245                           INC     -(R5)
 (2)  003110  000000                           HALT                            ; PROGRAM IS IN WRONG SEQUENCE
5490  003112  005215                    2$:    INC     (R5)
5491  003114  112703  000377                   MOVB    #377,R3                 ; LOAD REGISTER
5492  003120  112700  000252                   MOVB    #252,R0                 ; PLACE #252 IN R0
5493  003124  000277                           SCC
5494  003126  140003                           BICB    R0,R3                   ; CLEAR EVERY OTHER BIT
5495  003130  004737  017102                   JSR     PC,@#$CC1               ; CHECK FOR CC = 1
5496  003134  130003                           BITB    R0,R3                   ; CHECK IT
5497  003136  001404                           BEQ     4$                      ; CONTINUE IF OK
5498  003140  012745  000076                   MOV     #76,-(R5)
 (2)  003144  005245                           INC     -(R5)
 (2)  003146  000000                           HALT                            ; BICB OR BITB INSTRUCTION FAILED
5499  003150  132703  000125            4$:    BITB    #125,R3                 ; CHECK IT
5500  003154  004737  017102                   JSR     PC,@#$CC1               ; CHECK FOR CC = 1
5501  003160  150003                           BISB    R0,R3                   ; SET THE BITS THAT WERE CLEARED
5502  003162  100404                           BMI     6$
5503  003164  012745  000077                   MOV     #77,-(R5)
 (2)  003170  005245                           INC     -(R5)
 (2)  003172  000000                           HALT                            ; BISB INSTRUCTION FAILED
5504  003174  142703  000177            6$:    BICB    #177,R3                 ; CLEAR ALL THE BITS EXCEPT FOR SIGN
5505  003200  004737  017272                   JSR     PC,@#$CC11              ; CHECK FOR CC = 11
5506  003204  132703  000377                   BITB    #377,R3                 ; CHECK IT
5507  003210  004737  017272                   JSR     PC,@#$CC11              ; CHECK FOR CC = 11
5508
5509
5510
5511                                    ;;*************************************************************************
 (2)                                    ;*TEST: 21      NEW INSTRUCTIONS USED IN THIS SECTION ARE INCB, DECB
 (3)                                    ;;*************************************************************************
5512
5513  003214                            INCB0:
 (2)  003214  021527  000021                    CMP     (R5),#21
5514  003220  001404                           BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
5515  003222  012745  000100                   MOV     #100,-(R5)
 (2)  003226  005245                           INC     -(R5)
 (2)  003230  000000                           HALT                            ; PROGRAM IS IN WRONG SEQUENCE
5516  003232  005215                    1$:    INC     (R5)
5517  003234  112704  000177                   MOVB    #177,R4                 ; R4 = 177
5518  003240  000261                           SEC
5519  003242  105204                           INCB    R4                      ; ADD ONES INTO REG. 4
5520  003244  004737  017334                   JSR     PC,@#$CC13              ; CHECK FOR CC = 13
5521  003250  112704  000376                   MOVB    #376,R4
5522  003254  105204                           INCB    R4
5523  003256  004737  017272                   JSR     PC,@#$CC11              ; CHECK FOR CC = 11
5524  003262  105204                           INCB    R4
5525  003264  004737  017206                   JSR     PC,@#$CC5               ; CHECK FOR CC = 5
5526  003270  105204                           INCB    R4
```

```
 5527  003272  004737  017102                    JSR     PC,@#$CC1       ; CHECK FOR CC = 1
 5528  003276  122704  000001                    CMPB    #1,R4           ; CHECK IT
 5529  003302  001404                            BEQ     2$              ; CONTINUE IF OK
 5530  003304  012745  000101                    MOV     #101,-(R5)
   (2) 003310  005245                            INC     -(R5)
   (2) 003312  000000                            HALT                    ; INCB INSTRUCTION FAILED
 5531  003314  000261                2$:         SEC
 5532  003316  105304                            DECB    R4              ; SUBTRACT ONES FROM REG. 4
 5533  003320  004737  017206                    JSR     PC,@#$CC5       ; CHECK FOR CC = 5
 5534  003324  105304                            DECB    R4
 5535  003326  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 5536  003332  012704  000200                    MOV     #200,R4
 5537  003336  105304                            DECB    R4
 5538  003340  004737  017142                    JSR     PC,@#$CC3       ; CHECK FOR CC = 3
 5539  003344  105304                            DECB    R4
 5540  003346  004737  017102                    JSR     PC,@#$CC1       ; CHECK FOR CC = 1
```

```
5541                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 22      NEW INSTRUCTION IN THIS SECTION IS COMB
 (3)                                    ;;*******************************************************************************
5542
5543  003352                           COMBO:
 (2)  003352  021527  000022                    CMP     (R5),#22
5544  003356  001404                            BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
5545  003360  012745  000102                    MOV     #102,-(R5)
 (2)  003364  005245                            INC     -(R5)
 (2)  003366  000000                            HALT                    ; PROGRAM IS IN WRONG SEQUENCE
5546  003370  005215                    1$:     INC     (R5)
5547  003372  112703  000252                    MOVB    #252,R3         ; LOAD EVERY OTHER BIT
5548  003376  000277                            SCC
5549  003400  105103                            COMB    R3              ; 1'S COMPLEMENT
5550  003402  004737  017102                    JSR     PC,@#$CC1       ; CHECK FOR CC = 1
5551  003406  122703  000125                    CMPB    #125,R3         ; CHECK IT
5552  003412  001404                            BEQ     2$              ; CONTINUE IF OK
5553  003414  012745  000103                    MOV     #103,-(R5)
 (2)  003420  005245                            INC     -(R5)
 (2)  003422  000000                            HALT                    ; COMB INSTRUCTION FAILED
5554  003424  000277                    2$:     SCC
5555  003426  105103                            COMB    R3              ; COMPLEMENT BACK
5556  003430  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
5557  003434  122703  000252                    CMPB    #252,R3         ; CHECK IT
5558  003440  001404                            BEQ     3$              ; CONTINUE IF OK
5559  003442  012745  000104                    MOV     #104,-(R5)
 (2)  003446  005245                            INC     -(R5)
 (2)  003450  000000                            HALT                    ; COMB INSTRUCTION FAILED
5560  003452  012703  000377            3$:     MOV     #377,R3
5561  003456  000277                            SCC
5562  003460  105103                            COMB    R3
5563  003462  004737  017206                    JSR     PC,@#$CC5       ; CHECK FOR CC = 5
5564
5565
5566
5567                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 23      NEW INSTRUCTION IN THIS SECTION IS NEGB
 (3)                                    ;;*******************************************************************************
5568
5569  003466                           NEGBO:
 (2)  003466  021527  000023                    CMP     (R5),#23
5570  003472  001025                            BNE     ENEGBO          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
5571  003474  005215                    1$:     INC     (R5)
5572  003476  112700  000001                    MOVB    #1,R0           ; LOAD THE REGISTER
5573  003502  105400                            NEGB    R0              ; 2'S COMPLEMENT
5574  003504  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
5575  003510  122700  000377                    CMPB    #377,R0         ; CHECK IT
5576  003514  001404                            BEQ     2$              ; CONTINUE IF OK
5577  003516  012745  000105                    MOV     #105,-(R5)
 (2)  003522  005245                            INC     -(R5)
 (2)  003524  000000                            HALT                    ; NEGB INSTRUCTION FAILED
5578  003526  012700  000200            2$:     MOV     #200,R0
5579  003532  105400                            NEGB    R0              ; 2'S COMPLEMENT
5580  003534  004737  017334                    JSR     PC,@#$CC13      ; CHECK FOR CC = 13
5581  003540  122700  000200                    CMPB    #200,R0         ; CHECK IT
5582  003544  001404                            BEQ     ROLBO           ; CONTINUE IF OK
```

```
 5583  003546                          ENEGB0:
  (2)  003546  012745  000106                    MOV     #106,-(R5)
  (2)  003552  005245                            INC     -(R5)
  (2)  003554  000000                            HALT                    ; WRONG RESULT IN R0 OR WRONG SEQUENCE
```

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

003556                              ROLBO:
003556  021527  000024                      CMP     (R5),#24
003562  001026                              BNE     EROLBO          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
003564  005215                              INC     (R5)
003566  112701  000040                      MOVB    #40,R1          ; LOAD REGISTER
003572  000257                              CCC                     ; CLEAR FLAGS
003574  106101                              ROLB    R1              ; SHIFT
003576  106101                              ROLB    R1              ;
003600  004737  017314                      JSR     PC,@#$CC12      ; CHECK FOR CC = 12
003604  122701  000200                      CMPB    #200,R1         ; CHECK IT
003610  001404                              BEQ     1$              ; CONTINUE IF OK
003612  012745  000107                      MOV     #107,-(R5)
003616  005245                              INC     -(R5)
003620  000000                              HALT                    ; ROLB INSTRUCTION FAILED
003622  106101                      1$:     ROLB    R1              ; SHIFT
003624  004737  017230                      JSR     PC,@#$CC7       ; CHECK FOR CC = 7
003630  106101                              ROLB    R1              ; SHIFT
003632  122701  000001                      CMPB    #1,R1           ; CHECK IT
003636  001404                              BEQ     RORBO           ; CONTINUE IF OK
003640                              EROLBO:
003640  012745  000110                      MOV     #110,-(R5)
003644  005245                              INC     -(R5)
003646  000000                              HALT                    ; WRONG RESULT IN R1 OR WRONG SEQUENCE


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

003650                              RORBO:
003650  021527  000025                      CMP     (R5),#25
003654  001026                              BNE     ERORBO          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
003656  005215                              INC     (R5)
003660  112702  000004                      MOVB    #4,R2           ; LOAD REGISTER
003664  000257                              CCC                     ; CLEAR FLAGS
003666  106002                              RORB    R2              ; SHIFT
003670  106002                              RORB    R2              ;
003672  122702  000001                      CMPB    #1,R2           ; CHECK IT
003676  001404                              BEQ     1$              ; CONTINUE IF OK
003700  012745  000111                      MOV     #111,-(R5)
003704  005245                              INC     -(R5)
003706  000000                              HALT                    ; RORB INSTRUCTION FAILED
003710  106002                      1$:     RORB    R2              ; SHIFT
003712  004737  017230                      JSR     PC,@#$CC7       ; CHECK FOR CC = 7
003716  106002                              RORB    R2              ; SHIFT
003720  004737  017314                      JSR     PC,@#$CC12      ; CHECK FOR CC = 12
003724  122702  000200                      CMPB    #200,R2         ; CHECK IT
003730  001404                              BEQ     ASLBO           ; CONTINUE IF OK
003732                              ERORBO:
003732  012745  000112                      MOV     #112,-(R5)
003736  005245                              INC     -(R5)
003740  000000                              HALT
```

```
5624                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 26      NEW INSTRUCTION IN THIS SECTION IS ASLB
 (3)                                    ;;*******************************************************************************
5625
5626  003742                            ASLB0:
 (2)  003742  021527  000026                    CMP     (R5),#26
5627  003746  001404                            BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
5628  003750  012745  000113                    MOV     #113,-(R5)
 (2)  003754  005245                            INC     -(R5)
 (2)  003756  000000                            HALT                    ; PROGRAM IS IN WRONG SEQUENCE
5629  003760  005215                    2$:     INC     (R5)
5630  003762  112703  000040                    MOVB    #40,R3          ; LOAD REGISTER
5631  003766  000257                            CCC                     ; CLEAR FLAGS
5632  003770  106303                            ASLB    R3              ; SHIFT
5633  003772  106303                            ASLB    R3              ;
5634  003774  004737  017314                    JSR     PC,@#$CC12      ; CHECK FOR CC = 12
5635  004000  122703  000200                    CMPB    #200,R3         ; CHECK IT
5636  004004  001404                            BEQ     4$              ; CONTINUE IF OK
5637  004006  012745  000114                    MOV     #114,-(R5)
 (2)  004012  005245                            INC     -(R5)
 (2)  004014  000000                            HALT                    ; ASLB INSTRUCTION FAILED
5638  004016  106303                    4$:     ASLB    R3              ; SHIFT
5639  004020  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
5640  004024  106303                            ASLB    R3              ; SHIFT
5641  004026  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
5642
5643                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 27      NEW INSTRUCTION IN THIS SECTION IS ASRB
 (3)                                    ;;*******************************************************************************
5644
5645  004032                            ASRB0:
 (2)  004032  021527  000027                    CMP     (R5),#27
5646  004036  001034                            BNE     EASRB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
5647  004040  005215                    1$:     INC     (R5)
5648  004042  112704  000004                    MOVB    #4,R4           ; LOAD REGISTER
5649  004046  000257                            CCC                     ; CLEAR FLAGS
5650  004050  106204                            ASRB    R4              ; SHIFT
5651  004052  106204                            ASRB    R4              ;
5652  004054  122704  000001                    CMPB    #1,R4           ; CHECK IT
5653  004060  001404                            BEQ     2$              ; CONTINUE IF OK
5654  004062  012745  000115                    MOV     #115,-(R5)
 (2)  004066  005245                            INC     -(R5)
 (2)  004070  000000                            HALT                    ; ASRB INSTRUCTION FAILED
5655  004072  106204                    2$:     ASRB    R4              ; SHIFT
5656  004074  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
5657  004100  106204                            ASRB    R4              ; SHIFT
5658  004102  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
5659  004106  112703  000202                    MOVB    #202,R3         ; LOAD REGISTER
5660  004112  106203                            ASRB    R3              ; SHIFT
5661  004114  106203                            ASRB    R3              ;
5662  004116  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
5663  004122  122703  000340                    CMPB    #340,R3         ; CHECK IT
5664  004126  001404                            BEQ     ADCB0           ; CONTINUE IF OK
5665  004130                            EASRB0:
 (2)  004130  012745  000116                    MOV     #116,-(R5)
 (2)  004134  005245                            INC     -(R5)
```

```
    (2)  004136  000000                          HALT
```

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

004140                          ADCB0:
004140  021527  000030                  CMP     (R5),#30
004144  001404                          BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
004146  012745  000117                  MOV     #117,-(R5)
004152  005245                          INC     -(R5)
004154  000000                          HALT                            ; PROGRAM IS IN WRONG SEQUENCE
004156  005215                  2$:     INC     (R5)
004160  105000                          CLRB    R0                      ; CLEAR THE REGISTER
004162  000257                          CCC                             ; CLEAR FLAGS
004164  105500                          ADCB    R0                      ; ADD C BIT = 0
004166  004737  017164                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
004172  000261                          SEC                             ; C=1
004174  105500                          ADCB    R0                      ; ADD C BIT=1
004176  000261                          SEC                             ; C=1
004200  105500                          ADCB    R0                      ; AGAIN
004202  004737  017062                  JSR     PC,@#$CC0               ; CHECK FOR CC = 0
004206  122700  000002                  CMPB    #2,R0                   ; CHECK IT
004212  001404                          BEQ     4$                      ; CONTINUE IF OK
004214  012745  000120                  MOV     #120,-(R5)
004220  005245                          INC     -(R5)
004222  000000                          HALT                            ; ADCB INSTRUCTION FAILED
004224  112700  000177          4$:     MOVB    #177,R0                 ; LOAD LARGEST POSITVE NUMBER
004230  000261                          SEC                             ; C=1
004232  105500                          ADCB    R0                      ; ADD C BIT=1
004234  004737  017314                  JSR     PC,@#$CC12              ; CHECK FOR CC = 12
004240  122700  000200                  CMPB    #200,R0                 ; CHECK IT
004244  001404                          BEQ     6$                      ; CONTINUE IF OK
004246  012745  000121                  MOV     #121,-(R5)
004252  005245                          INC     -(R5)
004254  000000                          HALT                            ; ADCB INSTRUCTION FAILED
004256  112700  000377          6$:     MOVB    #377,R0                 ; LOAD -1
004262  000261                          SEC                             ; C=1
004264  105500                          ADCB    R0                      ; ADD C BIT=1
004266  004737  017206                  JSR     PC,@#$CC5               ; CHECK FOR CC = 5



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

004272                          SBCB0:
004272  021527  000031                  CMP     (R5),#31
004276  001404                          BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
004300  012745  000122                  MOV     #122,-(R5)
004304  005245                          INC     -(R5)
004306  000000                          HALT                            ; PROGRAM IS IN WRONG SEQUENCE
004310  005215                  1$:     INC     (R5)                    ; TEST IS IN WRONG SEQUENCE
004312  112701  000003                  MOVB    #3,R1                   ; LOAD REGISTER
004316  000257                          CCC                             ; CLEAR FLAGS
004320  105601                          SBCB    R1                      ; SUBTRACT C BIT=0
004322  004737  017062                  JSR     PC,@#$CC0               ; CHECK FOR CC = 0
```

```
 5708  004326  122701  000003                  CMPB    #3,R1           ; CHECK IT
 5709  004332  001404                          BEQ     2$              ; CONTINUE IF OK
 5710  004334  012745  000123                  MOV     #123,-(R5)
  (2)  004340  005245                          INC     -(R5)
  (2)  004342  000000                          HALT                    ; SBCB INSTRUCTION FAILED
 5711  004344  000261                  2$:     SEC                     ; C=1
 5712  004346  105601                          SBCB    R1              ; SUBTRACT C BIT=1
 5713  004350  000261                          SEC                     ; C=1
 5714  004352  105601                          SBCB    R1              ;
 5715  004354  004737  017062                  JSR     PC,@#$CC0       ; CHECK FOR CC = 0
 5716  004360  122701  000001                  CMPB    #1,R1           ; CHECK IT
 5717  004364  001404                          BEQ     3$              ; CONTINUE IF OK
 5718  004366  012745  000124                  MOV     #124,-(R5)
  (2)  004372  005245                          INC     -(R5)
  (2)  004374  000000                          HALT                    ; SBCB INSTRUCTION FAILED
 5719  004376  000261                  3$:     SEC                     ; C=1
 5720  004400  105601                          SBCB    R1              ; SUBTRACT C BIT=1
 5721  004402  004737  017164                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
 5722  004406  000261                          SEC                     ; C=1
 5723  004410  105601                          SBCB    R1              ; SUBTRACT C BIT = 1
 5724  004412  004737  017272                  JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 5725  004416  122701  000377                  CMPB    #377,R1         ; CHECK IT
 5726  004422  001404                          BEQ     4$              ; CONTINUE IF OK
 5727  004424  012745  000125                  MOV     #125,-(R5)
  (2)  004430  005245                          INC     -(R5)
  (2)  004432  000000                          HALT                    ; SBCB INSTRUCTION FAILED
 5728  004434  112701  000200          4$:     MOVB    #200,R1         ; LOAD R1
 5729  004440  000261                          SEC                     ; C=1
 5730  004442  105601                          SBCB    R1              ; SUBTRACT C BIT = 1
 5731  004444  004737  017122                  JSR     PC,@#$CC2       ; CHECK FOR CC = 2
```

```
                                        ;               CHECK WORD INSTRUCTIONS, DESTINATION MODE 0 ONLY
                                        ;               ----------------------------------------------
                                        ;;*************************************************************************
                                        ;*TEST: 32      NEW INSTRUCTIONS USED IN THIS SECTION ARE TST, CLR, MOV
                                        ;;*************************************************************************
004450                                  TSTO:
004450  021527  000032                          CMP     (R5),#32
004454  001404                                 BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
004456  012745  000126                          MOV     #126,-(R5)
004462  005245                                 INC     -(R5)
004464  000000                                 HALT                            ; TEST IS IN WRONG SEQUENCE
004466  005215                          1$:    INC     (R5)
004470  000277                                 SCC
004472  005000                                 CLR     R0                      ; CLEAR THE REGISTER
004474  004737  017164                         JSR     PC,@#$CC4               ; CHECK FOR CC = 4
004500  005700                                 TST     R0                      ; CHECK IT
004502  004737  017164                         JSR     PC,@#$CC4               ; CHECK FOR CC = 4
004506  012704  177777                         MOV     #177777,R4              ; LOAD THE REGISTER
004512  010401                                 MOV     R4,R1
004514  004737  017252                         JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004520  005701                                 TST     R1                      ; CHECK IT
004522  004737  017252                         JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004526  020401                                 CMP     R4,R1                   ; CHECK R1 TO CONTAIN PROPER DATA
004530  001404                                 BEQ     2$
004532  012745  000127                         MOV     #127,-(R5)
004536  005245                                 INC     -(R5)
004540  000000                                 HALT
004542  000263                          2$:    SEVC                            ; SET V & C BITS
004544  010000                                 MOV     R0,R0
004546  004767  012434                         JSR     PC,$CC5
                                        ;;*************************************************************************
                                        ;*TEST: 33      NEW INSTRUCTIONS USED IN THIS SECTION ARE CMP, BIS
                                        ;;*************************************************************************
004552                                  CMPO:
004552  021527  000033                          CMP     (R5),#33
004556  001026                                 BNE     ECMPO                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
004560  005215                          1$:    INC     (R5)
004562  012700  177777                         MOV     #177777,R0              ; LOAD REGISTER
004566  050002                                 BIS     R0,R2                   ; CHECK THE BIS INSTRUCTION
004570  004737  017252                         JSR     PC,@#$CC10              ; CHECK FOR CC = 10
004574  020002                                 CMP     R0,R2                   ; CHECK COMPARE
004576  001404                                 BEQ     2$                      ; CONTINUE IF OK
004600  012745  000130                         MOV     #130,-(R5)
004604  005245                                 INC     -(R5)
004606  000000                                 HALT                            ; BIS OR CMP INSTRUCTION FAILED
```

```
 5780  004610  022702  000077          2$:     CMP     #77,R2          ; CHECK IT AGAIN
 5781  004614  100004                          BPL     3$              ; CONTINUE IF OK
 5782  004616  012745  000131                  MOV     #131,-(R5)
   (2) 004622  005245                          INC     -(R5)
   (2) 004624  000000                          HALT                    ; CMP INSTRUCTION FAILED [WRONG CC]
 5783  004626  020227  000077          3$:     CMP     R2,#77          ; ONCE MORE
 5784  004632  100404                          BMI     BICO            ; CONTINUE IF OK
 5785  004634                          ECMPO:
   (2) 004634  012745  000132                  MOV     #132,-(R5)
   (2) 004640  005245                          INC     -(R5)
   (2) 004642  000000                          HALT                    ; WRONG CC OR WRONG SEQUENCE
```

```
5786                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 34      NEW INSTRUCTIONS USED IN THIS SECTION ARE BIC, BIT
 (3)                                    ;;*******************************************************************************
5787
5788  004644                            BICO:
 (2)  004644  021527  000034                    CMP     (R5),#34
5789  004650  001053                            BNE     EBICO           ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
5790  004652  005215                            INC     (R5)
5791  004654  012703  177777                    MOV     #177777,R3      ; LOAD REGISTER
5792  004660  012700  000440                    MOV     #TEMP,R0        ; PLACE THE ADDRESS OF LOCATION TEMP IN R0
5793  004664  012710  125252                    MOV     #125252,(R0)    ; SET (R0)
5794  004670  000277                            SCC
5795  004672  041003                            BIC     (R0),R3         ; CLEAR EVERY OTHER BIT
5796  004674  004737  017102                    JSR     PC,@#$CC1       ; CHECK FOR CC = 1
5797  004700  031003                            BIT     (R0),R3         ; CHECK IT
5798  004702  001404                            BEQ     1$              ; CONTINUE IF OK
5799  004704  012745  000133                    MOV     #133,-(R5)
 (2)  004710  005245                            INC     -(R5)
 (2)  004712  000000                            HALT                    ; BIC OR BIT INSTRUCTION FAILED
5800  004714  032703  052525            1$:     BIT     #52525,R3       ; CHECK IT
5801  004720  004737  017102                    JSR     PC,@#$CC1       ; CHECK FOR CC = 1
5802  004724  052703  125252                    BIS     #125252,R3      ; SET THE BITS THAT WERE CLEARED
5803  004730  100404                            BMI     2$              ; CONTINUE IF OK
5804  004732  012745  000134                    MOV     #134,-(R5)
 (2)  004736  005245                            INC     -(R5)
 (2)  004740  000000                            HALT                    ; BIT OR BIS INSTRUCTION FAILED
5805  004742  042703  077777            2$:     BIC     #77777,R3       ; CLEAR ALL THE BITS EXCEPT FOR SIGN
5806  004746  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
5807  004752  012700  177777                    MOV     #177777,R0
5808  004756  030003                            BIT     R0,R3           ; CHECK IT
5809  004760  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
5810  004764  000263                            SEVC                    ; SET V & C BITS
5811  004766  040000                            BIC     R0,R0
5812  004770  004737  017206                    JSR     PC,@#$CC5       ; CHECK CC = 5
5813  004774  005700                            TST     R0              ; CHECK R0 TO CONTAIN 0
5814  004776  001404                            BEQ     INCO
5815  005000                            EBICO:
 (2)  005000  012745  000135                    MOV     #135,-(R5)
 (2)  005004  005245                            INC     -(R5)
 (2)  005006  000000                            HALT                    ; BIC FAILED OR SEQUENCE ERROR
5816
5817
5818
5819                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 35      NEW INSTRUCTIONS USED IN THIS SECTION ARE INC, DEC
 (3)                                    ;;*******************************************************************************
5820
5821  005010                            INCO:
 (2)  005010  021527  000035                    CMP     (R5),#35
5822  005014  001404                            BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
5823  005016  012745  000136                    MOV     #136,-(R5)
 (2)  005022  005245                            INC     -(R5)
 (2)  005024  000000                            HALT                    ; PROGRAM IS IN WRONG SEQUENCE
5824  005026  005215            2$:     INC     (R5)
5825  005030  012704  077777                    MOV     #77777,R4       ; R4=77777
5826  005034  000261                            SEC
```

```
 5827  005036  005204                        INC     R4              ; ADD ONES INTO REG. 4
 5828  005040  004737  017334                JSR     PC,@#$CC13      ; CHECK FOR CC = 13
 5829  005044  012704  177776                MOV     #177776,R4
 5830  005050  005204                        INC     R4
 5831  005052  004737  017272                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 5832  005056  005204                        INC     R4
 5833  005060  004737  017206                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
 5834  005064  005204                        INC     R4
 5835  005066  004737  017102                JSR     PC,@#$CC1       ; CHECK FOR CC = 1
 5836  005072  022704  000001                CMP     #1,R4           ; CHECK IT
 5837  005076  001404                        BEQ     4$              ; FAILED
 5838  005100  012745  000137                MOV     #137,-(R5)
   (2) 005104  005245                        INC     -(R5)
   (2) 005106  000000                        HALT                    ; INC INSTRUCTION FAILED
 5839  005110  000261              4$:       SEC
 5840  005112  005304                        DEC     R4              ; SUBTRACT ONES FROM REG. 4
 5841  005114  004737  017206                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
 5842  005120  005304                        DEC     R4
 5843  005122  004737  017272                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 5844  005126  012704  100000                MOV     #100000,R4
 5845  005132  005304                        DEC     R4              ;
 5846  005134  004737  017142                JSR     PC,@#$CC3       ; CHECK FOR CC = 3
 5847  005140  005304                        DEC     R4              ;
 5848  005142  004737  017102                JSR     PC,@#$CC1       ; CHECK FOR CC = 1
```

```
5849                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 36      NEW INSTRUCTION IN THIS SECTION IS COM
 (3)                                    ;;*******************************************************************************
5850
5851  005146                           COMO:
 (2)  005146  021527  000036                CMP     (R5),#36
5852  005152  001404                          BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
5853  005154  012745  000140                MOV     #140,-(R5)
 (2)  005160  005245                          INC     -(R5)
 (2)  005162  000000                          HALT                    ; TEST IS IN WRONG SEQUENCE
5854  005164  005215                    1$:   INC     (R5)
5855  005166  012703  125252                MOV     #125252,R3      ; LOAD EVERY OTHER BIT
5856  005172  000277                          SCC
5857  005174  005103                          COM     R3              ; 1'S COMPLEMENT
5858  005176  004737  017102                JSR     PC,@#$CC1       ; CHECK FOR CC = 1
5859  005202  022703  052525                CMP     #52525,R3       ; CHECK IT
5860  005206  001404                          BEQ     2$              ; CONTINUE IF OK
5861  005210  012745  000141                MOV     #141,-(R5)
 (2)  005214  005245                          INC     -(R5)
 (2)  005216  000000                          HALT                    ; COM INSTRUCTION FAILED
5862  005220  000277                    2$:   SCC
5863  005222  005103                          COM     R3              ; COMPLEMENT BACK
5864  005224  004737  017272                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
5865  005230  022703  125252                CMP     #125252,R3      ; CHECK IT
5866  005234  001404                          BEQ     3$              ; CONTINUE IF OK
5867  005236  012745  000142                MOV     #142,-(R5)
 (2)  005242  005245                          INC     -(R5)
 (2)  005244  000000                          HALT                    ; COM INSTRUCTION FAILED
5868  005246  012703  177777          3$:   MOV     #177777,R3
5869  005252  000277                          SCC
5870  005254  005103                          COM     R3
5871  005256  004737  017206                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
5872
5873
5874
5875                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 37      NEW INSTRUCTION IN THIS SECTION IS NEG
 (3)                                    ;;*******************************************************************************
5876
5877  005262                           NEGO:
 (2)  005262  021527  000037                CMP     (R5),#37
5878  005266  001025                          BNE     ENEGO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
5879  005270  005215                    1$:   INC     (R5)
5880  005272  012700  000001                MOV     #1,R0           ; LOAD THE REGISTER
5881  005276  005400                          NEG     R0              ; 2'S COMPLEMENT
5882  005300  004737  017272                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
5883  005304  022700  177777                CMP     #177777,R0      ; CHECK IT
5884  005310  001404                          BEQ     2$              ; CONTINUE IF OK
5885  005312  012745  000143                MOV     #143,-(R5)
 (2)  005316  005245                          INC     -(R5)
 (2)  005320  000000                          HALT                    ; NEG INSTRUCTION FAILED
5886  005322  012700  100000          2$:   MOV     #100000,R0
5887  005326  005400                          NEG     R0              ; 2'S COMPLEMENT
5888  005330  004737  017334                JSR     PC,@#$CC13      ; CHECK FOR CC = 13
5889  005334  022700  100000                CMP     #100000,R0      ; CHECK IT
5890  005340  001404                          BEQ     ROLO            ; CONTINUE IF OK
```

```
DVKAAA  MACY11 27(732)  25-AUG-76  13:25  PAGE 54-33
DVKAAA.P11      T37     NEW INSTRUCTION IN THIS SECTION IS NEG                  *** SEQ 0036

  5891  005342                          ENEGO:
   (2)  005342  012745  000144                    MOV     #144,-(R5)
   (2)  005346  005245                            INC     -(R5)
   (2)  005350  000000                            HALT                    ; WRONG RESULT IN R0 OR WRONG SEQUENCE
```

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

005352                          ROLO:
005352  021527  000040                  CMP     (R5),#40
005356  001026                          BNE     EROLO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
005360  005215                          INC     (R5)
005362  012701  020000                  MOV     #20000,R1       ; LOAD REGISTER
005366  000257                          CCC                     ; CLEAR FLAGS
005370  006101                          ROL     R1              ; SHIFT
005372  006101                          ROL     R1              ;
005374  004737  017314                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
005400  022701  100000                  CMP     #100000,R1      ; CHECK IT
005404  001404                          BEQ     1$              ; CONTINUE IF OK
005406  012745  000145                  MOV     #145,-(R5)
005412  005245                          INC     -(R5)
005414  000000                          HALT                    ; ROL INSTRUCTION FAILED
005416  006101                  1$:     ROL     R1              ; SHIFT
005420  004737  017230                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
005424  006101                          ROL     R1              ; SHIFT
005426  022701  000001                  CMP     #1,R1           ; CHECK IT
005432  001404                          BEQ     RORO            ; CONTINUE IF OK
005434                          EROLO:
005434  012745  000146                  MOV     #146,-(R5)
005440  005245                          INC     -(R5)
005442  000000                          HALT                    ; WRONG RESULT IN R1 OR WRONG SEQUENCE



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

005444                          RORO:
005444  021527  000041                  CMP     (R5),#41
005450  001026                          BNE     ERORO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
005452  005215                          INC     (R5)
005454  012702  000004                  MOV     #4,R2           ; LOAD REGISTER
005460  000257                          CCC                     ; CLEAR FLAGS
005462  006002                          ROR     R2              ; SHIFT
005464  006002                          ROR     R2              ;
005466  022702  000001                  CMP     #1,R2           ; CHECK IT
005472  001404                          BEQ     1$              ; CONTINUE IF OK
005474  012745  000147                  MOV     #147,-(R5)
005500  005245                          INC     -(R5)
005502  000000                          HALT                    ; ROR INSTRUCTION FAILED
005504  006002                  1$:     ROR     R2              ; SHIFT
005506  004737  017230                  JSR     PC,@#$CC7       ; CHECK FOR CC = 7
005512  006002                          ROR     R2              ; SHIFT
005514  004737  017314                  JSR     PC,@#$CC12      ; CHECK FOR CC = 12
005520  022702  100000                  CMP     #100000,R2      ; CHECK IT
005524  001404                          BEQ     ASLO            ; CONTINUE IF OK
005526                          ERORO:
005526  012745  000150                  MOV     #150,-(R5)
005532  005245                          INC     -(R5)
```

```
   (2)  005534  000000                          HALT                    ; WRONG RESULT IN R2 OR WRONG SEQUENCE
```

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

005536                  ASLO:
005536  021527  000042          CMP     (R5),#42
005542  001404                  BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
005544  012745  000151          MOV     #151,-(R5)
005550  005245                  INC     -(R5)
005552  000000                  HALT                    ; PROGRAM IS IN WRONG SEQUENCE
005554  005215  2$:     INC     (R5)
005556  012703  020000          MOV     #20000,R3       ; LOAD REGISTER
005562  000257                  CCC                     ; CLEAR FLAGS
005564  006303                  ASL     R3              ; SHIFT
005566  006303                  ASL     R3              ;
005570  004737  017314          JSR     PC,@#$CC12      ; CHECK FOR CC = 12
005574  022703  100000          CMP     #100000,R3      ; CHECK IT
005600  001404                  BEQ     4$              ; CONTINUE IF OK
005602  012745  000152          MOV     #152,-(R5)
005606  005245                  INC     -(R5)
005610  000000                  HALT                    ; ASL INSTRUCTION FAILED
005612  006303  4$:     ASL     R3              ; SHIFT
005614  004737  017230          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
005620  006303                  ASL     R3              ; SHIFT
005622  004737  017164          JSR     PC,@#$CC4       ; CHECK FOR CC = 4

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

005626                  ASRO:
005626  021527  000043          CMP     (R5),#43
005632  001034                  BNE     EASRO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
005634  005215  1$:     INC     (R5)
005636  012704  000004          MOV     #4,R4           ; LOAD REGISTER
005642  000257                  CCC                     ; CLEAR FLAGS
005644  006204                  ASR     R4              ; SHIFT
005646  006204                  ASR     R4              ;
005650  022704  000001          CMP     #1,R4           ; CHECK IT
005654  001404                  BEQ     2$              ; CONTINUE IF OK
005656  012745  000153          MOV     #153,-(R5)
005662  005245                  INC     -(R5)
005664  000000                  HALT                    ; ASR INSTRUCTION FAILED
005666  006204  2$:     ASR     R4              ; SHIFT
005670  004737  017230          JSR     PC,@#$CC7       ; CHECK FOR CC = 7
005674  006204                  ASR     R4              ; SHIFT
005676  004737  017164          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
005702  012703  100002          MOV     #100002,R3      ; LOAD REGISTER
005706  006203                  ASR     R3              ; SHIFT
005710  006203                  ASR     R3              ;
005712  004737  017272          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
005716  022703  160000          CMP     #160000,R3      ; CHECK IT
005722  001404                  BEQ     ADCO            ; CONTINUE IF OK
005724  EASRO:
005724  012745  000154          MOV     #154,-(R5)
005730  005245                  INC     -(R5)
```

```
   (2)  005732  000000                    HALT                ; WRONG RESULT IN R3 OR WRONG SEQUENCE
```

```
5975                                    ;;*************************************************************************
 (2)                                    ;*TEST: 44      NEW INSTRUCTION IN THIS SECTION IS ADC
 (3)                                    ;;*************************************************************************
5976
5977  005734                            ADC0:
 (2)  005734  021527  000044                    CMP     (R5),#44
5978  005740  001404                            BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
5979  005742  012745  000155                    MOV     #155,-(R5)
 (2)  005746  005245                            INC     -(R5)
 (2)  005750  000000                            HALT                            ; PROGRAM IS IN WRONG SEQUENCE
5980  005752  005215                    2$:     INC     (R5)
5981  005754  005000                            CLR     R0                      ; CLEAR THE REGISTER
5982  005756  000257                            CCC                             ; CLEAR FLAGS
5983  005760  005500                            ADC     R0                      ; ADD C BIT = 0
5984  005762  004737  017164                    JSR     PC,@#$CC4               ; CHECK FOR CC = 4
5985  005766  000261                            SEC                             ; C=1
5986  005770  005500                            ADC     R0                      ; ADD C BIT=1
5987  005772  000261                            SEC                             ; C=1
5988  005774  005500                            ADC     R0                      ; AGAIN
5989  005776  004737  017062                    JSR     PC,@#$CC0               ; CHECK FOR CC = 0
5990  006002  022700  000002                    CMP     #2,R0                   ; CHECK IT
5991  006006  001404                            BEQ     4$                      ; CONTINUE IF OK
5992  006010  012745  000156                    MOV     #156,-(R5)
 (2)  006014  005245                            INC     -(R5)
 (2)  006016  000000                            HALT                            ; ADC INSTRUCTION FAILED
5993  006020  012700  077777            4$:     MOV     #77777,R0               ; LOAD LARGEST POSITVE NUMBER
5994  006024  000261                            SEC                             ; C=1
5995  006026  005500                            ADC     R0                      ; ADD C BIT=1
5996  006030  004737  017314                    JSR     PC,@#$CC12              ; CHECK FOR CC = 12
5997  006034  022700  100000                    CMP     #100000,R0              ; CHECK IT
5998  006040  001404                            BEQ     6$                      ; FAILED
5999  006042  012745  000157                    MOV     #157,-(R5)
 (2)  006046  005245                            INC     -(R5)
 (2)  006050  000000                            HALT                            ; ADC INSTRUCTION FAILED
6000  006052  012700  177777            6$:     MOV     #-1,R0                  ; LOAD -1
6001  006056  000261                            SEC                             ; C=1
6002  006060  005500                            ADC     R0                      ; ADD C BIT=1
6003  006062  004737  017206                    JSR     PC,@#$CC5               ; CHECK FOR CC = 5
6004
6005
6006
6007                                    ;;*************************************************************************
 (2)                                    ;*TEST: 45      NEW INSTRUCTION IN THIS SECTION IS SBC
 (3)                                    ;;*************************************************************************
6008
6009  006066                            SBC0:
 (2)  006066  021527  000045                    CMP     (R5),#45
6010  006072  001404                            BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
6011  006074  012745  000160                    MOV     #160,-(R5)
 (2)  006100  005245                            INC     -(R5)
 (2)  006102  000000                            HALT                            ; TEST IS IN WRONG SEQUENCE
6012  006104  005215                    1$:     INC     (R5)
6013  006106  012701  000003                    MOV     #3,R1                   ; LOAD REGISTER
6014  006112  000257                            CCC                             ; CLEAR FLAGS
6015  006114  005601                            SBC     R1                      ; SUBTRACT C BIT=0
6016  006116  004737  017062                    JSR     PC,@#$CC0               ; CHECK FOR CC = 0
```

```
 6017  006122  022701  000003                    CMP     #3,R1           ; CHECK IT
 6018  006126  001404                            BEQ     2$              ; CONTINUE IF OK
 6019  006130  012745  000161                    MOV     #161,-(R5)
   (2) 006134  005245                            INC     -(R5)
   (2) 006136  000000                            HALT                    ; SBC INSTRUCTION FAILED
 6020  006140  000261                  2$:       SEC                     ; C=1
 6021  006142  005601                            SBC     R1              ; SUBTRACT C BIT=1
 6022  006144  000261                            SEC                     ; C=1
 6023  006146  005601                            SBC     R1
 6024  006150  004737  017062                    JSR     PC,@#$CC0       ; CHECK FOR CC = 0
 6025  006154  022701  000001                    CMP     #1,R1           ; CHECK IT
 6026  006160  001404                            BEQ     3$              ; CONTINUE IF OK
 6027  006162  012745  000162                    MOV     #162,-(R5)
   (2) 006166  005245                            INC     -(R5)
   (2) 006170  000000                            HALT                    ; SBC INSTRUCTION FAILED
 6028  006172  000261                  3$:       SEC                     ; C=1
 6029  006174  005601                            SBC     R1              ; SUBTRACT C BIT=1
 6030  006176  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
 6031  006202  000261                            SEC                     ; C=1
 6032  006204  005601                            SBC     R1              ; SUBTRACT C BIT = 1
 6033  006206  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 6034  006212  022701  177777                    CMP     #-1,R1          ; CHECK IT
 6035  006216  001404                            BEQ     4$              ; CONTINUE IF F OK
 6036  006220  012745  000163                    MOV     #163,-(R5)
   (2) 006224  005245                            INC     -(R5)
   (2) 006226  000000                            HALT                    ; SBC INSTRUCTION FAILED
 6037  006230  012701  100000          4$:       MOV     #100000,R1      ; LOAD R1
 6038  006234  000261                            SEC                     ; C=1
 6039  006236  005601                            SBC     R1              ; SUBTRACT C BIT = 1
 6040  006240  004737  017122                    JSR     PC,@#$CC2       ; CHECK FOR CC = 2
```

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

006244                                  SXT0:
006244  021527  000046                          CMP     (R5),#46
006250  001024                                  BNE     ESXT0           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
006252  005215                          1$:     INC     (R5)
006254  005002                                  CLR     R2              ; CLEAR REGISTER
006256  000277                                  SCC
006260  000254                                  CLNZ
006262  006702                                  SXT     R2              ; SIGN EXTEND
006264  004737  017206                          JSR     PC,@#$CC5       ; CHECK FOR CC = 5
006270  005702                                  TST     R2              ; REG. 2 SHOULD STILL BE 0
006272  001404                                  BEQ     2$              ; CONTINUE IF OK
006274  012745  000164                          MOV     #164,-(R5)
006300  005245                                  INC     -(R5)
006302  000000                                  HALT                    ; SXT INSTRUCTION FAILED
006304  000273                          2$:     SENVC                   ; SET N, V & C BITS
006306  006702                                  SXT     R2              ; SIGN EXTEND
006310  004737  017272                          JSR     PC,@#$CC11      ; CHECK FOR CC = 11
006314  022702  177777                          CMP     #-1,R2          ; REG. 2 SHOULD NOW HAVE -1
006320  001404                                  BEQ     SWAB0           ; CONTINUE IF OK
006322                                  ESXT0:
006322  012745  000165                          MOV     #165,-(R5)
006326  005245                                  INC     -(R5)
006330  000000                                  HALT                    ; WRONG RESULT IN R2 OR WRONG SEQUENCE




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

006332                                  SWAB0:
006332  021527  000047                          CMP     (R5),#47
006336  001031                                  BNE     ESWAB0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
006340  005215                                  INC     (R5)
006342  012703  125125                          MOV     #125125,R3      ; LOAD BIT PATTERN INTO REGISTER
006346  000277                                  SCC
006350  000250                                  CLN
006352  000303                                  SWAB    R3              ; SWAP BYTES OF REGISTER
006354  004737  017252                          JSR     PC,@#$CC10      ; CHECK FOR CC = 10
006360  022703  052652                          CMP     #52652,R3       ; CHECK IT
006364  001404                                  BEQ     1$              ; CONTINUE IF OK
006366  012745  000166                          MOV     #166,-(R5)
006372  005245                                  INC     -(R5)
006374  000000                                  HALT                    ; SWAB INSTRUCTION FAILED
006376  012703  000377                  1$:     MOV     #377,R3
006402  000277                                  SCC
006404  000244                                  CLZ
006406  000303                                  SWAB    R3
006410  004737  017164                          JSR     PC,@#$CC4       ; CHECK FOR CC = 4
006414  022703  177400                          CMP     #177400,R3
006420  001404                                  BEQ     XOR0
006422                                  ESWAB0:
```

F04


```
    (2)  006422  012745  000167                    MOV     #167,-(R5)
    (2)  006426  005245                            INC     -(R5)
    (2)  006430  000000                            HALT                    ; WRONG RESULT IN R3 OR WRONG SEQUENCE
```

```
6084                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 50      NEW INSTRUCTION IN THIS SECTION IS XOR
 (3)                                    ;;*******************************************************************************
6085
6086  006432                            XORO:
 (2)  006432  021527  000050                    CMP     (R5),#50
6087  006436  001034                            BNE     EXORO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6088  006440  005215                            INC     (R5)
6089  006442  012704  177777                    MOV     #-1,R4          ; LOAD REGISTERS
6090  006446  012703  177777                    MOV     #-1,R3          ;
6091  006452  000277                            SCC
6092  006454  074403                            XOR     R4,R3           ; SHOULD PRODUCE 0'S IN REG. 3
6093  006456  004737  017206                    JSR     PC,@#$CC5       ; CHECK FOR CC = 5
6094  006462  012703  077777                    MOV     #77777,R3
6095  006466  010400                            MOV     R4,R0           ; PLACE A -1 IN R0
6096  006470  000263                            SEVC                    ; SET V & C BITS
6097  006472  000244                            CLZ
6098  006474  074003                            XOR     R0,R3
6099  006476  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
6100  006502  012702  125252                    MOV     #125252,R2      ; LOAD REGISTERS
6101  006506  012704  052525                    MOV     #52525,R4       ;
6102  006512  000277                            SCC
6103  006514  074204                            XOR     R2,R4           ; SHOULD PRODUCE ALL 1'S IN REG. 4
6104  006516  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
6105  006522  022704  177777                    CMP     #-1,R4          ; CHECK IT
6106  006526  001404                            BEQ     ADDO            ; CONTINUE IF OK
6107  006530                            EXORO:
 (2)  006530  012745  000170                    MOV     #170,-(R5)
 (2)  006534  005245                            INC     -(R5)
 (2)  006536  000000                            HALT                    ; WRONG RESULT IN R4 OR WRONG SEQUENCE
6108
6109
6110
6111                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 51      NEW INSTRUCTION IN THIS SECTION IS ADD
 (3)                                    ;;*******************************************************************************
6112
6113  006540                            ADDO:
 (2)  006540  021527  000051                    CMP     (R5),#51
6114  006544  001055                            BNE     EADDO           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6115  006546  005215                            INC     (R5)
6116  006550  012701  021421                    MOV     #21421,R1       ; LOAD REGISTERS
6117  006554  060101                            ADD     R1,R1           ; ADD
6118  006556  004737  017062                    JSR     PC,@#$CCO       ; CHECK FOR CC = 0
6119  006562  022701  043042                    CMP     #43042,R1       ; CHECK IT
6120  006566  001404                            BEQ     1$              ; CONTINUE IF OK
6121  006570  012745  000171                    MOV     #171,-(R5)
 (2)  006574  005245                            INC     -(R5)
 (2)  006576  000000                            HALT                    ; ADD INSTRUCTION FAILED
6122  006600  012700  156357          1$:       MOV     #-21421,R0      ; LOAD REGISTERS
6123  006604  060000                            ADD     R0,R0           ; ADD
6124  006606  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
6125  006612  022700  134736                    CMP     #-43042,R0      ; CHECK IT
6126  006616  001404                            BEQ     2$              ; CONTINUE IF OK
6127  006620  012745  000172                    MOV     #172,-(R5)
 (2)  006624  005245                            INC     -(R5)
```

```
  (2)  006626  000000                        HALT
                                             ; ADD INSTRUCTION FAILED
6128   006630  012702  100000      2$:       MOV     #100000,R2
                                             ; LOAD REGISTERS
6129   006634  060202                        ADD     R2,R2
                                             ; ADD SHOULD RESULT AS 0'S
6130   006636  004737  017230                JSR     PC,@#$CC7
                                             ;CHECK FOR CC = 7
6131   006642  012704  021421                MOV     #21421,R4
                                             ; LOAD REGISTERS
6132   006646  012701  156357                MOV     #-21421,R1
                                             ;
6133   006652  060401                        ADD     R4,R1
                                             ; ADD SHOULD RESULT AS 0'S
6134   006654  001404                        BEQ     3$
                                             ; CONTINUE IF OK
6135   006656  012745  000173                MOV     #173,-(R5)
  (2)  006662  005245                        INC     -(R5)
  (2)  006664  000000                        HALT
                                             ; ADD INSTRUCTION FAILED
6136   006666  005404              3$:       NEG     R4
                                             ; SWITCH SOURCE AND DESTINATION
6137   006670  012701  021421                MOV     #21421,R1
                                             ;
6138   006674  060104                        ADD     R1,R4
                                             ; SHOULD RESULT AS 0'S
6139   006676  001404                        BEQ     SUB0
                                             ; CONTINUE IF OK
6140   006700                      EADD0:
  (2)  006700  012745  000174                MOV     #174,-(R5)
  (2)  006704  005245                        INC     -(R5)
  (2)  006706  000000                        HALT
                                             ; WRONG RESULT IN R1 OR WRONG SEQUENCE
```

```
                                ;;*******************************************************************************
                                ;*TEST: 52      NEW INSTRUCTION IN THIS SECTION IS SUB
                                ;;*******************************************************************************
                        SUBO:
006710  021527  000052                CMP     (R5),#52
006714  001404                        BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
006716  012745  000175                MOV     #175,-(R5)
006722  005245                        INC     -(R5)
006724  000000                        HALT                    ; PROGRAM IS IN WRONG SEQUENCE
006726  005215                2$:     INC     (R5)
006730  012702  021421                MOV     #21421,R2       ; LOAD REGISTERS
006734  012703  156357                MOV     #-21421,R3      ;
006740  160203                        SUB     R2,R3           ; RESULT SHOULD=-43042
006742  004737  017252                JSR     PC,@#$CC10      ; CHECK FOR CC = 10
006746  022703  134736                CMP     #-43042,R3      ; CHECK IT
006752  001404                        BEQ     4$              ; CONTINUE IF OK
006754  012745  000176                MOV     #176,-(R5)
006760  005245                        INC     -(R5)
006762  000000                        HALT                    ; SUB INSTRUCTION FAILED
006764  012703  021421        4$:     MOV     #21421,R3       ; LOAD REGISTER
006770  010204                        MOV     R2,R4           ; NOW R4 = #21421
006772  160403                        SUB     R4,R3           ; RESULT SHOULD=0
006774  001404                        BEQ     6$
006776  012745  000177                MOV     #177,-(R5)
007002  005245                        INC     -(R5)
007004  000000                        HALT                    ; SUB NSTRUCTION FAILED
007006  012703  177777        6$:     MOV     #-1,R3          ; LOAD REGISTERS
007012  012702  077777                MOV     #77777,R2       ; LOAD REGISTERS
007016  160302                        SUB     R3,R2           ; RESULT SHOULD BE 100000 AND OVERFLOW
007020  004737  017334                JSR     PC,@#$CC13      ; CHECK FOR CC = 13
007024  022702  100000                CMP     #100000,R2      ; CHECK IT
007030  001404                        BEQ     8$              ; CONTINUE IF OK
007032  012745  000200                MOV     #200,-(R5)
007036  005245                        INC     -(R5)
007040  000000                        HALT                    ; SUB INSTRUCTION FAILED
007042  012704  177777        8$:     MOV     #-1,R4
007046  160304                        SUB     R3,R4
007050  004737  017164                JSR     PC,@#$CC4       ; CHECK FOR CC = 4


                                ;;*******************************************************************************
                                ;*TEST: 53      NEW INSTRUCTIONS IN THARE SECTION IS MTPS & MFPS
                                ;;*******************************************************************************
                        PSW:
007054  021527  000053                CMP     (R5),#53
007060  001032                        BNE     EPSW            ; IF IN WRONG SEQUENCE THEN GO TO HLT AT THE END OF THE
007062  005215                1$:     INC     (R5)
007064  012701  177777                MOV     #177777,R1
007070  005000                        CLR     R0
007072                                MTPS    R0              ; SET PSW TO 0
007072  106400                        .WORD   106400!..C
007074  004737  017062                JSR     PC,@#$CC0       ; CHECK FOR CC = 0
007100                                MFPS    R1              ; MOVE PSW TO R1
007100  106701                        .WORD   106700!..C
```

```
 6187  007102  001404                          BEQ     2$                ; CONTINUE IF BIT 8 OF PSW WAS EXTENDED IN R1
 6188  007104  012745  000201                  MOV     #201,-(R5)
   (2) 007110  005245                          INC     -(R5)
   (2) 007112  000000                          HALT                      ; MTPS OR MFPS INSTRUCTION FAILED
 6189  007114  004737  017164        2$:       JSR     PC,@#$CC4         ; CHECK FOR CC = 4
 6190  007120  012700  000377                  MOV     #377,R0
 6194  007124                                  MTPS    R0                ; SET PSW TO 357 SINCE MTPS DOES NOT SET T BIT
   (1) 007124  106400                          .WORD   106400!..C
 6195  007126  004737  017354                  JSR     PC,@#$CC17        ; CHECK FOR CC = 17
 6196  007132                                  MFPS    R1                ; MOVE PSW TO R1
   (1) 007132  106701                          .WORD   106700!..C
 6200  007134  004737  017272                  JSR     PC,@#$CC11        ; CHECK FOR CC = 11 IC BIT SHOULD NOT BE EFFECTED BY MFP
 6201  007140  022701  177757                  CMP     #177757,R1        ; CHECK TO SEE IF BIT 8 OF PSW WAS EXTENDED THRU R1
 6202  007144  001404                          BEQ     MODE0
 6203  007146                          EPSW:
   (2) 007146  012745  000202                  MOV     #202,-(R5)
   (2) 007152  005245                          INC     -(R5)
   (2) 007154  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
                                        ;;*************************************************************************
007156                                  MODE0:
  (2)  007156  021527  000054                   CMP     (R5),#54
007162  001063                                  BNE     EMODE0          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
007164  005215                                  INC     (R5)
007166  112700  000252                         MOVB    #252,R0         ; LOAD REGISTERS
007172  110001                                  MOVB    R0,R1           ;
007174  110102                                  MOVB    R1,R2           ;
007176  122702  000252                         CMPB    #252,R2         ; CHECK IT
007202  001404                                  BEQ     1$              ; OK, CONTINUE
007204  012745  000203                         MOV     #203,-(R5)
  (2)  007210  005245                           INC     -(R5)
  (2)  007212  000000                           HALT                    ; MOV INSTRUCTION FAILED IN MODE 0
007214  012700  125252                 1$:     MOV     #125252,R0      ; LOAD REGISTERS
007220  010001                                  MOV     R0,R1           ;
007222  010102                                  MOV     R1,R2           ;
007224  022702  125252                         CMP     #125252,R2      ; CHECK IT
007230  001404                                  BEQ     MODE1           ; OK, CONTINUE
007232  012745  000204                         MOV     #204,-(R5)
  (2)  007236  005245                           INC     -(R5)
  (2)  007240  000000                           HALT                    ; MOV INSTRUCTION FAILED IN MODE 0

007242  012700  000440                 MODE1:  MOV     #TEMP,R0        ; LOAD ADDRESSES INTO REGS.
007246  012701  000442                         MOV     #TEMP1,R1       ;
007252  012702  000444                         MOV     #TEMP2,R2       ;
007256  005067  171162                         CLR     TEMP2           ; START CLEAN
007262  112710  000125                         MOVB    #125,(R0)       ; LOAD THE LOCATIONS
007266  111011                                  MOVB    (R0),(R1)       ; TEMP ---> TEMP1
007270  111112                                  MOVB    (R1),(R2)       ; TEMP1 ---> TEMP2
007272  122767  000125  171144                 CMPB    #125,TEMP2      ; CHECK IT
007300  001404                                  BEQ     1$              ; OK, CONTINUE
007302  012745  000205                         MOV     #205,-(R5)
  (2)  007306  005245                           INC     -(R5)
  (2)  007310  000000                           HALT                    ; MOV INSTRUCTION FAILED IN MODE 1
007312  012710  052525                 1$:     MOV     #52525,(R0)     ; LOAD THE LOCATIONS
007316  011011                                  MOV     (R0),(R1)       ; TEMP ---> TEMP1
007320  011112                                  MOV     (R1),(R2)       ; TEMP1 ---> TEMP2
007322  022767  052525  171114                 CMP     #52525,TEMP2    ; CHECK IT
007330  001404                                  BEQ     MODE2           ; OK, CONTINUE
007332                                  EMODE0:
  (2)  007332  012745  000206                   MOV     #206,-(R5)
  (2)  007336  005245                           INC     -(R5)
  (2)  007340  000000                           HALT                    ; MOV INSTRUCTION FAILED IN MODE 1
                                                                        ; OR WRONG SEQUENCE

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

```
  (2)                                   ;*TEST: 55      CHECK MODE 2 USING THE MOVB AND MOV INSTRUCTIONS
  (3)                                   ;;*************************************************************************
 6252
 6253  007342                         MODE2:
  (2)  007342  021527  000055                 CMP     (R5),#55
 6254  007346  001050                       BNE     EMODE2          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
 6255  007350  005215                       INC     (R5)
 6256  007352  012700  000440               MOV     #TEMP,R0        ; LOAD ADDRESSES
 6257  007356  012701  000442               MOV     #TEMP1,R1       ;
 6258  007362  012702  000444               MOV     #TEMP2,R2
 6259  007366  105022                       CLRB    (R2)+           ; START CLEAN
 6260  007370  112710  000252               MOVB    #252,(R0)       ; LOAD THE LOCATIONS
 6261  007374  112021                       MOVB    (R0)+,(R1)+     ; TEMP ---> TEMP1
 6262  007376  105201                       INCB    R1              ; MAKE IT EVEN
 6263  007400  111167  171034               MOVB    (R1),TEMP       ; MORE 0'S INTO TEMP
 6264  007404  105200                       INCB    R0              ; MAKE IT EVEN
 6265  007406  112021                       MOVB    (R0)+,(R1)+     ; TEMP1 ---> TEMP2
 6266  007410  124227  000252               CMPB    -(R2),#252      ; CHECK IT
 6267  007414  001003                       BNE     1$              ; FAILED
 6268  007416  105767  171016               TSTB    TEMP            ; CHECK IT
 6269  007422  001404                       BEQ     2$              ; OK, CONTINUE
 6270  007424                         1$:
  (2)  007424  012745  000207               MOV     #207,-(R5)
  (2)  007430  005245                       INC     -(R5)
  (2)  007432  000000                       HALT                    ; INSTRUCTIONS FAILED IN MODE 2
 6271
 6272  007434  005741                 2$:   TST     -(R1)
 6273  007436  005022                       CLR     (R2)+           ; START CLEAN
 6274  007440  012740  125252               MOV     #125252,-(R0)   ; LOAD LOCATIONS
 6275  007444  012020                       MOV     (R0)+,(R0)+     ; TEMP ---> TEMP1
 6276  007446  011067  170766               MOV     (R0),TEMP       ; 0 ---> TEMP
 6277  007452  012121                       MOV     (R1)+,(R1)+     ; 125252 ---> TEMP2
 6278  007454  024227  125252               CMP     -(R2),#125252   ; CHECK IT
 6279  007460  001003                       BNE     EMODE2          ; FAILED
 6280  007462  005767  170752               TST     TEMP            ; CHECK IT
 6281  007466  001404                       BEQ     MODE3           ; OK, CONTINUE
 6282  007470                         EMODE2:
  (2)  007470  012745  000210               MOV     #210,-(R5)
  (2)  007474  005245                       INC     -(R5)
  (2)  007476  000000                       HALT                    ; INSTRUCTIONS FAILED IN MODE 2
 6283                                                               ; OR WRONG SEQUENCE
```

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

        007500                          MODE3:
        007500  021527  000056                  CMP     (R5),#56
        007504  001066                          BNE     EMODE3                  ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
        007506  005215                          INC     (R5)
        007510  012767  000440  170712          MOV     #TEMP,ADR               ; LOAD ADDRESSES
        007516  012767  000442  170706          MOV     #TEMP1,ADR1             ;
        007524  012767  000444  170702          MOV     #TEMP2,ADR2             ;
        007532  012700  000430                  MOV     #ADR,R0                 ; LOAD ADDRESSES OF ADDRESSES
        007536  012701  000432                  MOV     #ADR1,R1                ;
        007542  105067  170676                  CLRB    TEMP2                   ; START CLEAN
        007546  112767  000125  170664          MOVB    #125,TEMP               ;
        007554  113031                          MOVB    @(R0)+,@(R1)+           ; TEMP ---> TEMP1
        007556  113167  170656                  MOVB    @(R1)+,TEMP             ; TEMP2 ---> TEMP
        007562  113030                          MOVB    @(R0)+,@(R0)+           ; TEMP1 ---> TEMP2
        007564  122767  000125  170652          CMPB    #125,TEMP2              ; CHECK IT
        007572  001003                          BNE     1$                      ; FAILED
        007574  105767  170640                  TSTB    TEMP                    ; CHECK IT
        007600  001404                          BEQ     2$                      ; OK, CONTINUE
        007602                          1$:
        007602  012745  000211                  MOV     #211,-(R5)
        007606  005245                          INC     -(R5)
        007610  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 3
        007612  005067  170626          2$:     CLR     TEMP2                   ; START CLEAN
        007616  012767  052525  170614          MOV     #52525,TEMP             ; LOAD LOCATIONS
        007624  012700  000430                  MOV     #ADR,R0                 ; LOAD ADDRESSES OF ADDRESSES
        007630  012701  000432                  MOV     #ADR1,R1                ;
        007634  013030                          MOV     @(R0)+,@(R0)+           ; TEMP ---> TEMP1
        007636  013067  170576                  MOV     @(R0)+,TEMP             ; TEMP2 ---> TEMP
        007642  013131                          MOV     @(R1)+,@(R1)+           ; TEMP1 ---> TEMP2
        007644  022767  052525  170572          CMP     #52525,TEMP2            ; CHECK IT
        007652  001003                          BNE     EMODE3                  ; FAILED
        007654  005767  170560                  TST     TEMP                    ; CHECK IT
        007660  001404                          BEQ     MODE4                   ; OK, CONTINUE
        007662                          EMODE3:
        007662  012745  000212                  MOV     #212,-(R5)
        007666  005245                          INC     -(R5)
        007670  000000                          HALT                            ; INSTRUCTIONS FAILED IN MODE 3




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

        007672                          MODE4:
        007672  021527  000057                  CMP     (R5),#57
        007676  001120                          BNE     EMODE4                  ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
        007700  005215                          INC     (R5)
        007702  105067  170532                  CLRB    TEMP                    ; START CLEAN
        007706  012700  000440                  MOV     #TEMP,R0                ; LOAD ADDRESSES
        007712  012701  000442                  MOV     #TEMP1,R1               ;
        007716  012702  000444                  MOV     #TEMP2,R2               ;
```

```
 6328  007722  005202                        INC     R2                      ; ADJUST THE POINTER
 6329  007724  021267  170515                CMP     (R2),TEMP2+1
 6330  007730  001404                        BEQ     1$
 6331  007732  012745  000213                MOV     #213,-(R5)
   (2) 007736  005245                        INC     -(R5)
   (2) 007740  000000                        HALT                            ; INSTRUCTIONS FAILED IN MODE 4
 6332  007742  112742  000252        1$:     MOVB    #252,-(R2)              ; LOAD TEMP2
 6333  007746  005201                        INC     R1                      ; ADJUST THE POINTERS
 6334  007750  005202                        INC     R2                      ;
 6335  007752  114241                        MOVB    -(R2),-(R1)             ; TEMP2 ---> TEMP1
 6336  007754  005200                        INC     R0                      ; ADJUST THE POINTERS
 6337  007756  005202                        INC     R2                      ;
 6338  007760  114042                        MOVB    -(R0),-(R2)             ; TEMP ---> TEMP2
 6339  007762  105200                        INCB    R0                      ; ADJUST THE POINTERS
 6340  007764  021067  170451                CMP     (R0),TEMP+1
 6341  007770  001404                        BEQ     2$
 6342  007772  012745  000214                MOV     #214,-(R5)
   (2) 007776  005245                        INC     -(R5)
   (2) 010000  000000                        HALT                            ; INSTRUCTIONS FAILED IN MODE 4
 6343  010002  105201            2$:         INCB    R1                      ;
 6344  010004  114140                        MOVB    -(R1),-(R0)             ; TEMP1 ---> TEMP
 6345  010006  122767  000252  170424        CMPB    #252,TEMP               ; CHECK IT
 6346  010014  001003                        BNE     3$                      ; FAILED
 6347  010016  105767  170422                TSTB    TEMP2                   ; CHECK IT
 6348  010022  001404                        BEQ     4$                      ; OK, CONTINUE
 6349  010024                    3$:
   (2) 010024  012745  000215                MOV     #215,-(R5)
   (2) 010030  005245                        INC     -(R5)
   (2) 010032  000000                        HALT                            ; INSTRUCTIONS FAILED IN MODE 4
 6350  010034  005067  170400        4$:     CLR     TEMP                    ; START CLEAN
 6351  010040  012700  000440                MOV     #TEMP,R0                ; LOAD ADDRESSES
 6352  010044  012701  000442                MOV     #TEMP1,R1               ;
 6353  010050  012702  000444                MOV     #TEMP2,R2               ;
 6354  010054  005722                        TST     (R2)+                   ; ADJUST THE POINTER
 6355  010056  021267  170364                CMP     (R2),TEMP2+2
 6356  010062  001404                        BEQ     5$
 6357  010064  012745  000216                MOV     #216,-(R5)
   (2) 010070  005245                        INC     -(R5)
   (2) 010072  000000                        HALT                            ; INSTRUCTIONS FAILED IN MODE 4
 6358  010074  012742  125252        5$:     MOV     #125252,-(R2)           ; LOAD TEMP2
 6359  010100  005721                        TST     (R1)+                   ; ADJUST THE POINTERS
 6360  010102  005722                        TST     (R2)+                   ;
 6361  010104  014241                        MOV     -(R2),-(R1)             ; TEMP2 ---> TEMP1
 6362  010106  005720                        TST     (R0)+                   ; ADJUST POINTERS
 6363  010110  005722                        TST     (R2)+                   ;
 6364  010112  014042                        MOV     -(R0),-(R2)             ; TEMP ---> TEMP2
 6365  010114  005720                        TST     (R0)+                   ; ADJUST THE POINTERS
 6366  010116  005721                        TST     (R1)+                   ;
 6367  010120  014140                        MOV     -(R1),-(R0)             ; TEMP1 ---> TEMP
 6368  010122  022767  125252  170310        CMP     #125252,TEMP            ; CHECK IT
 6369  010130  001003                        BNE     EMODE4                  ; FAILED
 6370  010132  005767  170306                TST     TEMP2                   ; CHECK IT
 6371  010136  001404                        BEQ     MODE5                   ; OK, CONTINUE
 6372  010140                    EMODE4:
   (2) 010140  012745  000217                MOV     #217,-(R5)
   (2) 010144  005245                        INC     -(R5)
```

```
   (2)  010146  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 4
 6373                                                                    ; OR WRONG SEQUENCE
```

```
6374                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 60      CHECK MODE 5 USING THE MOVB AND MOV INSTRUCTIONS
 (3)                                    ;;*******************************************************************************
6375
6376  010150                            MODE5:
 (2)  010150  021527  000060                    CMP     (R5),#60
6377  010154  001105                            BNE     EMODE5          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6378  010156  005215                            INC     (R5)
6379  010160  105067  170254                    CLRB    TEMP            ; START CLEAN
6380  010164  012767  000440  170236            MOV     #TEMP,ADR       ; LOAD ADDRESSES
6381  010172  012767  000442  170232            MOV     #TEMP1,ADR1     ;
6382  010200  012767  000444  170226            MOV     #TEMP2,ADR2     ;
6383  010206  012700  000430                    MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
6384  010212  012701  000432                    MOV     #ADR1,R1        ;
6385  010216  012702  000434                    MOV     #ADR2,R2        ;
6386  010222  005722                            TST     (R2)+           ; ADJUST THE POINTER
6387  010224  112752  000125                    MOVB    #125,@-(R2)     ; LOAD TEMP2
6388  010230  022122                            CMP     (R1)+,(R2)+     ; ADJUST THE POINTERS
6389  010232  115251                            MOVB    @-(R2),@-(R1)   ; TEMP2 ---> TEMP1
6390  010234  022022                            CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
6391  010236  115052                            MOVB    @-(R0),@-(R2)   ; TEMP ---> TEMP2
6392  010240  022022                            CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
6393  010242  125052                            CMPB    @-(R0),@-(R2)   ;CHECK IT
6394  010244  001404                            BEQ     1$
6395  010246  012745  000220                    MOV     #220,-(R5)
 (2)  010252  005245                            INC     -(R5)
 (2)  010254  000000                            HALT
6396  010256  022120                    1$:     CMP     (R1)+,(R0)+     ; ADJUST THE POINTERS
6397  010260  115150                            MOVB    @-(R1),@-(R0)   ; TEMP1 ---> TEMP
6398  010262  122767  000125  170150            CMPB    #125,TEMP       ; CHECK IT
6399  010270  001003                            BNE     2$              ; FAILED
6400  010272  105767  170146                    TSTB    TEMP2           ; CHECK IT
6401  010276  001404                            BEQ     3$              ; OK, CONTINUE
6402  010300                            2$:
 (2)  010300  012745  000221                    MOV     #221,-(R5)
 (2)  010304  005245                            INC     -(R5)
 (2)  010306  000000                            HALT                    ; INSTRUCTIONS FAILED IN MODE 5
6403  010310  005067  170124            3$:     CLR     TEMP            ; START CLEAN
6404  010314  012700  000430                    MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
6405  010320  012701  000432                    MOV     #ADR1,R1        ;
6406  010324  012702  000434                    MOV     #ADR2,R2        ;
6407  010330  005722                            TST     (R2)+           ; ADJUST THE POINTER
6408  010332  012752  052525                    MOV     #52525,@-(R2)   ; LOAD TEMP2
6409  010336  022122                            CMP     (R1)+,(R2)+     ; ADJUST THE POINTERS
6410  010340  015251                            MOV     @-(R2),@-(R1)   ; TEMP2 ---> TEMP1
6411  010342  022022                            CMP     (R0)+,(R2)+     ; ADJUST THE POINTERS
6412  010344  015052                            MOV     @-(R0),@-(R2)   ; TEMP ---> TEMP2
6413  010346  022021                            CMP     (R0)+,(R1)+     ; ADJUST THE POINTERS
6414  010350  015150                            MOV     @-(R1),@-(R0)   ; TEMP1 ---> TEMP
6415  010352  022767  052525  170060            CMP     #52525,TEMP     ; CHECK IT
6416  010360  001003                            BNE     EMODE5          ; FAILED
6417  010362  005767  170056                    TST     TEMP2           ; CHECK IT
6418  010366  001404                            BEQ     MODE6           ; OK, CONTINUE
6419  010370                            EMODE5:
 (2)  010370  012745  000222                    MOV     #222,-(R5)
 (2)  010374  005245                            INC     -(R5)
```

```
010376  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 5
                                                                ; OR WRONG SEQUENCE



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

010400                          MODE6:
010400  021527  000061                  CMP     (R5),#61
010404  001055                          BNE     EMODE6          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
010406  005215                          INC     (R5)
010410  005067  170030                  CLR     TEMP2           ; START CLEAN
010414  012700  000440                  MOV     #TEMP,R0        ; LOAD ADDRESSES
010420  012701  000442                  MOV     #TEMP1,R1       ;
010424  012702  000444                  MOV     #TEMP2,R2       ;
010430  112760  000252  000000          MOVB    #252,0(R0)      ; LOAD TEMP (LOW BYTE)
010436  112760  000252  000001          MOVB    #252,1(R0)      ; LOAD TEMP (HIGH BYTE)
010444  022767  125252  167766          CMP     #125252,TEMP    ; CHECK IT
010452  001012                          BNE     1$              ; FAILED
010454  116062  000001  000000          MOVB    1(R0),0(R2)     ; TEMP(H) ---> TEMP2(L)
010462  116160  000002  000005          MOVB    2(R1),5(R0)     ; TEMP2(L) ---> TEMP2(H)
010470  022767  125252  167746          CMP     #125252,TEMP2   ; CHECK IT
010476  001404                          BEQ     2$              ; OK, CONTINUE
010500                          1$:
010500  012745  000223                  MOV     #223,-(R5)
010504  005245                          INC     -(R5)
010506  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 6
010510  005067  167726          2$:     CLR     TEMP1           ; START CLEAN
010514  012760  052525  000000          MOV     #52525,0(R0)    ; LOAD TEMP
010522  016260  177774  000002          MOV     -4(R2),2(R0)    ; TEMP ---> TEMP1
010530  022767  052525  167704          CMP     #52525,TEMP1    ; CHECK IT
010536  001404                          BEQ     MODE7           ; OK, CONTINUE
010540                          EMODE6:
010540  012745  000224                  MOV     #224,-(R5)
010544  005245                          INC     -(R5)
010546  000000                          HALT                    ; INSTRUCTIONS FAILED IN MODE 6
                                                                ; OR WRONG SEQUENCE
```

```
6448                                            ;;*****************************************************************
 (2)                                            ;*TEST: 62      CHECK MODE 7 USING THE MOVB AND MOV INSTRUCTIONS
 (3)                                            ;;*****************************************************************
6449
6450  010550                                    MODE7:
 (2)  010550  021527  000062                            CMP     (R5),#62
6451  010554  001052                                    BNE     EMODE7          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6452  010556  005215                                    INC     (R5)
6453  010560  005067  167656                            CLR     TEMP1           ; START CLEAN
6454  010564  012767  000440  167636                    MOV     #TEMP,ADR       ; LOAD ADDRESSES
6455  010572  012767  000442  167632                    MOV     #TEMP1,ADR1     ;
6456  010600  012767  000444  167626                    MOV     #TEMP2,ADR2     ;
6457  010606  012700  000430                            MOV     #ADR,R0         ; LOAD ADDRESSES OF ADDRESSES
6458  010612  012701  000432                            MOV     #ADR1,R1        ;
6459  010616  012702  000434                            MOV     #ADR2,R2        ;
6460  010622  112770  000252  000000                    MOVB    #252,@0(R0)     ; LOAD TEMP
6461  010630  117270  177774  000002                    MOVB    @-4(R2),@2(R0)  ; TEMP ---> TEMP1
6462  010636  122767  000252  167576                    CMPB    #252,TEMP1      ; CHECK IT
6463  010644  001404                                    BEQ     1$              ; OK, CONTINUE
6464  010646  012745  000225                            MOV     #225,-(R5)
 (2)  010652  005245                                    INC     -(R5)
 (2)  010654  000000                                    HALT                    ; MODE 7 IS FAILING
6465  010656  012770  125252  000000            1$:     MOV     #125252,@0(R0)  ; LOAD TEMP
6466  010664  017270  177774  000002                    MOV     @-4(R2),@2(R0)  ; TEMP ---> TEMP1
6467  010672  022767  125252  167542                    CMP     #125252,TEMP1   ; CHECK IT
6468  010700  001404                                    BEQ     TSTB1           ; OK, CONTINUE
6469  010702                                    EMODE7:
 (2)  010702  012745  000226                            MOV     #226,-(R5)
 (2)  010706  005245                                    INC     -(R5)
 (2)  010710  000000                                    HALT                    ; INSTRUCTIONS FAILED IN MODE 7
6470                                                                            ; OR WRONG SEQUENCE
```

```
                                        ;       CHECK BYTE INSTRUCTIONS, NOT DESTINATION MODE 0
                                        ;       -----------------------------------------------
                                        ;;*************************************************************************
                                        ;*TEST: 63      NEW INSTRUCTIONS USED IN THIS SECTION ARE TSTB, CLRB, MOVB
                                        ;;*************************************************************************
                                        TSTB1:
 010712                                         CMP     (R5),#63
 010712  021527  000063                 (2)
 010716  001042                                 BNE     ETSTB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF TEST
 010720  005215                         2$:     INC     (R5)
 010722  012700  000440                         MOV     #TEMP,R0        ; LOAD ADDRESSES
 010726  012701  000442                         MOV     #TEMP1,R1       ;
 010732  000277                                 SCC
 010734  105010                                 CLRB    (R0)            ; CLEAR THE LOCATION
 010736  004737  017164                         JSR     PC,@#$CC4       ; CHECK FOR CC = 4
 010742  105710                                 TSTB    (R0)            ; CHECK IT
 010744  004737  017164                         JSR     PC,@#$CC4       ; CHECK FOR CC = 4
 010750  112711  000377                         MOVB    #377,(R1)       ; LOAD THE LOCATION
 010754  004737  017252                         JSR     PC,@#$CC10      ; CHECK FOR CC = 10
 010760  105711                                 TSTB    (R1)            ; CHECK IT
 010762  004737  017252                         JSR     PC,@#$CC10      ; CHECK FOR CC = 10
 010766  010002                                 MOV     R0,R2           ; R2 IS NOW POINTING TO LOCATION TEMP
 010770  112762  000200  000000                 MOVB    #200,0(R2)      ; PLACE #200 IN LOCATION TEMP
 010776  112241                                 MOVB    (R2)+,-(R1)     ; MOVE #200 TO LOCATION TEMP+1
 011000  026127  177777  100200                 CMP     -1(R1),#100200  ; CHECK THE DATA IN LOCATION TEMP
 011006  001404                                 BEQ     4$
 011010  012745  000227                         MOV     #227,-(R5)
 011014  005245                 (2)             INC     -(R5)
 011016  000000                 (2)             HALT                    ; MOVB INSTRUCTION FAILED
 011020  020102                         4$:     CMP     R1,R2           ; CHECK THE REGISTERS FOR PROPER VALUE
 011022  001404                                 BEQ     CMPB1
 011024                                 ETSTB1:
 011024  012745  000230                 (2)             MOV     #230,-(R5)
 011030  005245                 (2)             INC     -(R5)
 011032  000000                 (2)             HALT                    ; MOVB INSTRUCTION FAILED OR WRONG SEQUENCE
```

```
6509                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 64      NEW INSTRUCTIONS USED IN THIS SECTION ARE CMPB, BISB
 (3)                                    ;;*******************************************************************************
6510
6511  011034                           CMPB1:
 (2)  011034  021527  000064                    CMP     (R5),#64
6512  011040  001032                           BNE     ECMPB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6513  011042  005215                    1$:     INC     (R5)
6514  011044  012701  000444                    MOV     #TEMP2,R1
6515  011050  012702  000440                    MOV     #TEMP,R2        ; LOAD ADDRESS
6516  011054  012711  000077                    MOV     #77,(R1)        ; PLACE 77 IN LOCATION TEMP2
6517  011060  112704  000377                    MOVB    #377,R4         ; R4 SHOULD CONTAIN #177777
6518  011064  150412                           BISB    R4,(R2)         ; LOAD LOCATION
6519  011066  004737  017252                    JSR     PC,@#$CC10      ; CHECK FOR CC = 10
6520  011072  120412                           CMPB    R4,(R2)         ; CHECK COMPARE
6521  011074  001404                           BEQ     2$              ; CONTINUE IF OK
6522  011076  012745  000231                    MOV     #231,-(R5)
 (2)  011102  005245                           INC     -(R5)
 (2)  011104  000000                           HALT                    ; BISB OR CMPB INSTRUCTION FAILED
6523  011106  121112                    2$:     CMPB    (R1),(R2)       ; CHECK IT AGAIN
6524  011110  100004                           BPL     3$              ; CONTINUE IF OK
6525  011112  012745  000232                    MOV     #232,-(R5)
 (2)  011116  005245                           INC     -(R5)
 (2)  011120  000000                           HALT                    ; CMPB INSTRUCTION FAILED [WRONG CC]
6526  011122  121211                    3$:     CMPB    (R2),(R1)       ; ONCE MORE
6527  011124  100404                           BMI     BICB1           ; CONTINUE IF OK
6528  011126                           ECMPB1:
 (2)  011126  012745  000233                    MOV     #233,-(R5)
 (2)  011132  005245                           INC     -(R5)
 (2)  011134  000000                           HALT
6529
6530
6531
6532                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 65      NEW INSTRUCTIONS USED IN THIS SECTION ARE BICB, BITB
 (3)                                    ;;*******************************************************************************
6533
6534  011136                           BICB1:
 (2)  011136  021527  000065                    CMP     (R5),#65
6535  011142  001404                           BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
6536  011144  012745  000234                    MOV     #234,-(R5)
 (2)  011150  005245                           INC     -(R5)
 (2)  011152  000000                           HALT                    ; PROGRAM IS IN WRONG SEQUENCE
6537  011154  005215                    2$:     INC     (R5)
6538  011156  012703  000440                    MOV     #TEMP,R3        ; LOAD ADDRESS
6539  011162  112713  000377                    MOVB    #377,(R3)       ; LOAD LOCATION
6540  011166  012700  000442                    MOV     #TEMP1,R0       ;PLACE THE ADDRESS OF LOCATION TEMP1 IN R0
6541  011172  010001                           MOV     R0,R1           ;AND R1
6542  011174  112721  000252                    MOVB    #252,(R1)+      ; PLACE #252 IN TEMP1
6543  011200  000277                           SCC
6544  011202  146013  000000                    BICB    0(R0),(R3)      ; CLEAR EVERY OTHER BIT
6545  011206  004737  017102                    JSR     PC,@#$CC1       ; CHECK FOR CC = 1
6546  011212  136113  177777                    BITB    -1(R1),(R3)     ; CHECK IT
6547  011216  001404                           BEQ     4$              ; CONTINUE IF OK
6548  011220  012745  000235                    MOV     #235,-(R5)
 (2)  011224  005245                           INC     -(R5)
```

```
  (2) 011226 000000                      HALT                    ; BICB OR BITB INSTRUCTION FAILED
6549 011230 132713 000125          4$:   BITB   #125,(R3)        ; CHECK IT
6550 011234 004737 017102                JSR    PC,@#$CC1       ; CHECK FOR CC = 1
6551 011240 154113                       BISB   -(R1),(R3)      ; SET THE BITS THAT WERE CLEARED
6552 011242 100404                       BMI    6$              ; CONTINUE IF OK
6553 011244 012745 000236                MOV    #236,-(R5)
  (2) 011250 005245                      INC    -(R5)
  (2) 011252 000000                      HALT                    ; BITB OR BISB INSTRUCTION FAILED
6554 011254 012746 000177          6$:   MOV    #177,-(SP)      ; STORE #177 ON THE STACK
6555 011260 142613                       BICB   (SP)+,(R3)      ; CLEAR ALL THE BITS EXCEPT SIGN BIT
6556 011262 004737 017272                JSR    PC,@#$CC11      ; CHECK FOR CC = 11
6557 011266 132713 000377                BITB   #377,(R3)       ; CHECK IT
6558 011272 004737 017272                JSR    PC,@#$CC11      ; CHECK FOR CC = 11
6559 011276 010300                       MOV    R3,R0           ; PLACE THE ADDRESS OF LOCATION TEMP IN R0
6560 011300 012710 000442                MOV    #TEMP1,(R0)     ; PLACE THE ADDRESS OF LOCATION TEMP1 IN TEMP
6561 011304 012730 000377                MOV    #377,@(R0)+     ; WRITE A 377 IN LOCATION TEMP1
6562 011310 000263                       SEVC                   ; SET V & C BITS
6563 011312 145070 000000                BICB   @-(R0),@(R0)    ; BIT CLEAR THE CONTENTS
6564                                                            ; OF TEMP1 TO THE CONTENTS OF TEMP1
6565 011316 004737 017206                JSR    PC,@#$CC5       ; CHECK FOR CC = 5
6566 011322 022027 000442                CMP    (R0)+,#TEMP1    ; MAKE SURE THAT (R0) IS POINTING TO LOCATION TEMP1
6567 011326 001404                       BEQ    8$
6568 011330 012745 000237                MOV    #237,-(R5)
  (2) 011334 005245                      INC    -(R5)
  (2) 011336 000000                      HALT                    ; BICB OR CMP INSTRUCTION FAILED IN THE SPECIFIC MODE
6569 011340 005750                 8$:   TST    @-(R0)          ; TEST LOCATION TEMP1
6570 011342 001404                       BEQ    10$
6571 011344 012745 000240                MOV    #240,-(R5)
  (2) 011350 005245                      INC    -(R5)
  (2) 011352 000000                      HALT                    ; BICB INSTRUCTION FAILED
6572 011354 000257                 10$:  CCC
6573 011356 141010                       BICB   (R0),(R0)       ; CLEAR THE LOCATION TEMP
6574 011360 004737 017164                JSR    PC,@#$CC4       ; CHECK FOR CC = 4
```

```
6575                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 66      NEW INSTRUCTIONS USED IN THIS SECTION ARE INCB, DECB
 (3)                                    ;;*******************************************************************************
6576
6577  011364                           INCB1:
 (2)  011364  021527  000066                    CMP     (R5),#66
6578  011370  001067                           BNE     EINCB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6579  011372  005215                    1$:    INC     (R5)
6580  011374  012704  000440                   MOV     #TEMP,R4        ; LOAD ADDRESS
6581  011400  112714  000177                   MOVB    #177,(R4)       ; TEMP LOCATION=177
6582  011404  000261                           SEC
6583  011406  105214                           INCB    (R4)            ; ADD ONES INTO LOCATION
6584  011410  004737  017334                   JSR     PC,@#$CC13      ; CHECK FOR CC = 13
6585  011414  012714  000376                   MOV     #376,(R4)
6586  011420  012700  017272                   MOV     #$CC11,R0       ; MAKE R0 POINT TO CHECKING ROUTINE FOR CC = 11
6587  011424  105224                           INCB    (R4)+           ;
6588  011426  004720                           JSR     PC,(R0)+        ; CHECK FOR CC = 11
6589  011430  105744                           TSTB    -(R4)           ; DECREMENT R4 BY 1
6590  011432  005746                           TST     -(SP)           ; AND SP BY 2
6591  011434  010426                           MOV     R4,(SP)+        ; PLACE THE ADDRESS OF TEMP ON THE STACK
6592  011436  000241                           CLC                     ; CLEAR C BIT
6593  011440  105256                           INCB    @-(SP)          ; INCREMENT THE CONTENTS OF LOCATION TEMP
6594  011442  004737  017164                   JSR     PC,@#$CC4       ; CHECK FOR CC = 4
6595  011446  123634                           CMPB    @(SP)+,@(R4)+   ; RESTORE STACK POINTER
6596  011450  000261                           SEC                     ; SET C BIT
6597  011452  105264  177777                   INCB    -1(R4)
6598  011456  004737  017102                   JSR     PC,@#$CC1       ; CHECK FOR CC = 1
6599  011462  124427  000001                   CMPB    -(R4),#1        ; CHECK IT
6600  011466  001404                           BEQ     2$              ; CONTINUE IF OK
6601  011470  012745  000241                   MOV     #241,-(R5)
 (2)  011474  005245                           INC     -(R5)
 (2)  011476  000000                           HALT                    ; INCB INSTRUCTION FAILED
6602  011500  000261                    2$:    SEC
6603  011502  105314                           DECB    (R4)            ; SUBTRACT ONES FROM LOCATION
6604  011504  004737  017206                   JSR     PC,@#$CC5       ; CHECK FOR CC = 5
6605  011510  105324                           DECB    (R4)+
6606  011512  004740                           JSR     PC,-(R0)        ; CHECK FOR CC = 11
6607  011514  112764  000200  177777           MOVB    #200,-1(R4)
6608  011522  105344                           DECB    -(R4)           ;
6609  011524  004760  177650                   JSR     PC,$CC3-$CC11(R0); CHECK FOR CC = 3
6610  011530  105364  000000                   DECB    0(R4)           ;
6611  011534  004737  017102                   JSR     PC,@#$CC1       ; CHECK FOR CC = 1
6612  011540  126427  000000  000176           CMPB    0(R4),#176
6613  011546  001404                           BEQ     COMB1
6614  011550                           EINCB1:
 (2)  011550  012745  000242                   MOV     #242,-(R5)
 (2)  011554  005245                           INC     -(R5)
 (2)  011556  000000                           HALT                    ; DECB INSTRUCTION FAILED OR SEQUENCE ERROR
6615
6616
6617
6618
6619
6620
6621
6622
```

```
                                        ;;*****************************************************************
  (2)                                   ;*TEST: 67      NEW INSTRUCTION IN THIS SECTION IS COMB
  (3)                                   ;;*****************************************************************

      011560                            COMB1:
  (2) 011560  021527  000067                    CMP     (R5),#67
      011564  001404                            BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
      011566  012745  000243                    MOV     #243,-(R5)
  (2) 011572  005245                            INC     -(R5)
  (2) 011574  000000                            HALT                            ; TEST IS IN WRONG SEQUENCE
      011576  005215                    1$:     INC     (R5)
      011600  012703  000440                    MOV     #TEMP,R3                ; LOAD ADDRESS
      011604  012704  000442                    MOV     #TEMP1,R4
      011610  012714  000252                    MOV     #252,(R4)
      011614  112413                            MOVB    (R4)+,(R3)              ; LOAD EVERY OTHER BIT
      011616  000277                            SCC
      011620  105113                            COMB    (R3)                    ; 1'S COMPLEMENT
      011622  004737  017102                    JSR     PC,@#$CC1               ; CHECK FOR CC = 1
      011626  122713  000125                    CMPB    #125,(R3)               ; CHECK IT
      011632  001404                            BEQ     2$                      ; CONTINUE IF OK
      011634  012745  000244                    MOV     #244,-(R5)
  (2) 011640  005245                            INC     -(R5)
  (2) 011642  000000                            HALT                            ; COMB INSTRUCTION FAILED
      011644  000277                    2$:     SCC
      011646  105113                            COMB    (R3)                    ; COMPLEMENT BACK
      011650  004737  017272                    JSR     PC,@#$CC11              ; CHECK FOR CC = 11
      011654  010400                            MOV     R4,R0
      011656  126013  177777                    CMPB    -1(R0),(R3)             ; CHECK IT
      011662  001404                            BEQ     3$                      ; CONTINUE IF OK
      011664  012745  000245                    MOV     #245,-(R5)
  (2) 011670  005245                            INC     -(R5)
  (2) 011672  000000                            HALT                            ; COMB INSTRUCTION FAILED
      011674  112724  000377            3$:     MOVB    #377,(R4)+
      011700  114413                            MOVB    -(R4),(R3)              ; PLACE #377 IN (R3)
      011702  000277                            SCC
      011704  105113                            COMB    (R3)
      011706  004737  017206                    JSR     PC,@#$CC5               ; CHECK FOR CC = 5
```

```
6652                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 70      NEW INSTRUCTION IN THIS SECTION IS NEGB
 (3)                                    ;;*******************************************************************************
6653
6654  011712                            NEGB1:
 (2)  011712  021527  000070                    CMP     (R5),#70
6655  011716  001027                            BNE     ENEGB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6656  011720  005215                    1$:     INC     (R5)
6657  011722  012700  000440                    MOV     #TEMP,R0        ; LOAD ADDRESS
6658  011726  112710  000001                    MOVB    #1,(R0)         ; LOAD THE LOCATION
6659  011732  105410                            NEGB    (R0)            ; 2'S COMPLEMENT
6660  011734  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
6661  011740  122710  000377                    CMPB    #377,(R0)       ; CHECK IT
6662  011744  001404                            BEQ     2$              ; CONTINUE IF OK
6663  011746  012745  000246                    MOV     #246,-(R5)
 (2)  011752  005245                            INC     -(R5)
 (2)  011754  000000                            HALT                    ; NEGB INSTRUCTION FAILED
6664  011756  012710  000200            2$:     MOV     #200,(R0)
6665  011762  105410                            NEGB    (R0)            ; 2'S COMPLEMENT
6666  011764  004737  017334                    JSR     PC,@#$CC13      ; CHECK FOR CC = 13
6667  011770  122710  000200                    CMPB    #200,(R0)       ; CHECK IT
6668  011774  001404                            BEQ     ROLB1           ; CONTINUE IF OK
6669  011776                            ENEGB1:
 (2)  011776  012745  000247                    MOV     #247,-(R5)
 (2)  012002  005245                            INC     -(R5)
 (2)  012004  000000                            HALT                    ; WRONG RESULT AT TEMP OR WRONG SEQUENCE
6670
6671
6672
6673                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 71      NEW INSTRUCTION IN THIS SECTION IS ROLB
 (3)                                    ;;*******************************************************************************
6674
6675  012006                            ROLB1:
 (2)  012006  021527  000071                    CMP     (R5),#71
6676  012012  001030                            BNE     EROLB1          ; IF IN WRONG SEQUENCE GO TO HLT ABOVE
6677  012014  005215                            INC     (R5)
6678  012016  012701  000442                    MOV     #TEMP1,R1       ; LOAD ADDRESS
6679  012022  112711  000040                    MOVB    #40,(R1)        ; LOAD LOCATION
6680  012026  000257                            CCC                     ; CLEAR FLAGS
6681  012030  106111                            ROLB    (R1)            ; SHIFT
6682  012032  106111                            ROLB    (R1)            ;
6683  012034  004737  017314                    JSR     PC,@#$CC12      ; CHECK FOR CC = 12
6684  012040  122711  000200                    CMPB    #200,(R1)       ; CHECK IT
6685  012044  001404                            BEQ     1$              ; CONTINUE IF OK
6686  012046  012745  000250                    MOV     #250,-(R5)
 (2)  012052  005245                            INC     -(R5)
 (2)  012054  000000                            HALT                    ; ROLB INSTRUCTION FAILED
6687  012056  106111                    1$:     ROLB    (R1)            ; SHIFT
6688  012060  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
6689  012064  106111                            ROLB    (R1)            ; SHIFT
6690  012066  122711  000001                    CMPB    #1,(R1)         ; CHECK IT
6691  012072  001404                            BEQ     RORB1           ; CONTINUE IF OK
6692  012074                            EROLB1:
 (2)  012074  012745  000251                    MOV     #251,-(R5)
 (2)  012100  005245                            INC     -(R5)
```

```
(2)  012102  000000              HALT                ; WRONG RESULT AT TEMP1 OR WRONG SEQUENCE
```

```
6693                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 72      NEW INSTRUCTION IN THIS SECTION IS RORB
 (3)                                    ;;*******************************************************************************
6694
6695  012104                            RORB1:
 (2)  012104  021527  000072                    CMP     (R5),#72
6696  012110  001030                            BNE     ERORB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6697  012112  005215                            INC     (R5)
6698  012114  012702  000442                    MOV     #TEMP1,R2       ; LOAD ADDRESS
6699  012120  112712  000004                    MOVB    #4,(R2)         ; LOAD LOCATION
6700  012124  000257                            CCC                     ; CLEAR FLAGS
6701  012126  106012                            RORB    (R2)            ; SHIFT
6702  012130  106012                            RORB    (R2)            ;
6703  012132  122712  000001                    CMPB    #1,(R2)         ; CHECK IT
6704  012136  001404                            BEQ     1$              ; CONTINUE IF OK
6705  012140  012745  000252                    MOV     #252,-(R5)
 (2)  012144  005245                            INC     -(R5)
 (2)  012146  000000                            HALT                    ; RORB INSTRUCTION FAILED
6706  012150  106012                    1$:     RORB    (R2)            ; SHIFT
6707  012152  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
6708  012156  106012                            RORB    (R2)            ; SHIFT
6709  012160  004737  017314                    JSR     PC,@#$CC12      ; CHECK FOR CC = 12
6710  012164  122712  000200                    CMPB    #200,(R2)       ; CHECK IT
6711  012170  001404                            BEQ     ASLB1           ; CONTINUE IF OK
6712  012172                            ERORB1:
 (2)  012172  012745  000253                    MOV     #253,-(R5)
 (2)  012176  005245                            INC     -(R5)
 (2)  012200  000000                            HALT                    ; WRONG RESULT AT TEMP1 OR WRONG SEQUENCE
6713
6714
6715
6716                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 73      NEW INSTRUCTION IN THIS SECTION IS ASLB
 (3)                                    ;;*******************************************************************************
6717
6718  012202                            ASLB1:
 (2)  012202  021527  000073                    CMP     (R5),#73
6719  012206  001404                            BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
6720  012210  012745  000254                    MOV     #254,-(R5)
 (2)  012214  005245                            INC     -(R5)
 (2)  012216  000000                            HALT                    ; PROGRAM IS IN WRONG SEQUENCE
6721  012220  005215                    2$:     INC     (R5)
6722  012222  012703  000442                    MOV     #TEMP1,R3       ; LOAD ADDRESS
6723  012226  112713  000040                    MOVB    #40,(R3)        ; LOAD LOCATION
6724  012232  000257                            CCC                     ; CLEAR FLAGS
6725  012234  106313                            ASLB    (R3)            ; SHIFT
6726  012236  106313                            ASLB    (R3)            ;
6727  012240  004737  017314                    JSR     PC,@#$CC12      ; CHECK FOR CC = 12
6728  012244  122713  000200                    CMPB    #200,(R3)       ; CHECK IT
6729  012250  001404                            BEQ     4$              ; CONTINUE IF OK
6730  012252  012745  000255                    MOV     #255,-(R5)
 (2)  012256  005245                            INC     -(R5)
 (2)  012260  000000                            HALT                    ; ASLB INSTRUCTION FAILED
6731  012262  106313                    4$:     ASLB    (R3)            ; SHIFT
6732  012264  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
6733  012270  106313                            ASLB    (R3)            ; SHIFT
```

NO5


```
  6734  012272  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
```

```
6735                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 74      NEW INSTRUCTION IN THIS SECTION IS ASRB
 (3)                                    ;;*******************************************************************************
6736
6737  012276                            ASRB1:
 (2)  012276  021527  000074                    CMP     (R5),#74
6738  012302  001040                            BNE     EASRB1          ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6739  012304  005215                    1$:     INC     (R5)
6740  012306  012704  000442                    MOV     #TEMP1,R4       ; LOAD ADDRESSES
6741  012312  012703  000444                    MOV     #TEMP2,R3       ;
6742  012316  112714  000004                    MOVB    #4,(R4)         ; LOAD LOCATION
6743  012322  000257                            CCC                     ; CLEAR FLAGS
6744  012324  106214                            ASRB    (R4)            ; SHIFT
6745  012326  106214                            ASRB    (R4)            ;
6746  012330  122714  000001                    CMPB    #1,(R4)         ; CHECK IT
6747  012334  001404                            BEQ     2$              ; CONTINUE IF OK
6748  012336  012745  000256                    MOV     #256,-(R5)
 (2)  012342  005245                            INC     -(R5)
 (2)  012344  000000                            HALT                    ; ASRB INSTRUCTION FAILED
6749  012346  106214                    2$:     ASRB    (R4)            ; SHIFT
6750  012350  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
6751  012354  106214                            ASRB    (R4)            ; SHIFT
6752  012356  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
6753  012362  112713  000202                    MOVB    #202,(R3)       ; LOAD LOCATION
6754  012366  106213                            ASRB    (R3)            ; SHIFT
6755  012370  106213                            ASRB    (R3)            ;
6756  012372  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
6757  012376  122713  000340                    CMPB    #340,(R3)       ; CHECK IT
6758  012402  001404                            BEQ     ADCB1           ; CONTINUE IF OK
6759  012404                            EASRB1:
 (2)  012404  012745  000257                    MOV     #257,-(R5)
 (2)  012410  005245                            INC     -(R5)
 (2)  012412  000000                            HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE
6760
6761
6762
6763                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 75      NEW INSTRUCTION IN THIS SECTION IS ADCB
 (3)                                    ;;*******************************************************************************
6764
6765  012414                            ADCB1:
 (2)  012414  021527  000075                    CMP     (R5),#75
6766  012420  001404                            BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
6767  012422  012745  000260                    MOV     #260,-(R5)
 (2)  012426  005245                            INC     -(R5)
 (2)  012430  000000                            HALT                    ; PROGRAM IS IN WRONG SEQUENCE
6768  012432  005215                    2$:     INC     (R5)
6769  012434  012700  000444                    MOV     #TEMP2,R0       ; LOAD ADDRESS
6770  012440  105010                            CLRB    (R0)            ; CLEAR THE LOCATION
6771  012442  000257                            CCC                     ; CLEAR FLAGS
6772  012444  105510                            ADCB    (R0)            ; ADD C BIT = 0
6773  012446  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
6774  012452  000261                            SEC                     ; C=1
6775  012454  105510                            ADCB    (R0)            ; ADD C BIT=1
6776  012456  000261                            SEC                     ; C=1
6777  012460  105510                            ADCB    (R0)            ; AGAIN
```

```
 6778  012462  004737  017062                JSR     PC,@#$CC0       ; CHECK FOR CC = 0
 6779  012466  122710  000002                CMPB    #2,(R0)         ; CHECK IT
 6780  012472  001404                        BEQ     4$              ; CONTINUE IF OK
 6781  012474  012745  000261                MOV     #261,-(R5)
   (2) 012500  005245                        INC     -(R5)
   (2) 012502  000000                        HALT                    ; ADCB INSTRUCTION FAILED
 6782  012504  112710  000177          4$:   MOVB    #177,(R0)       ; LOAD LARGEST POSITVE BYTE
 6783  012510  000261                        SEC                     ; C=1
 6784  012512  105510                        ADCB    (R0)            ; ADD C BIT=1
 6785  012514  004737  017314                JSR     PC,@#$CC12      ; CHECK FOR CC = 12
 6786  012520  122710  000200                CMPB    #200,(R0)       ; CHECK IT
 6787  012524  001404                        BEQ     6$              ; CONTINUE IF OK
 6788  012526  012745  000262                MOV     #262,-(R5)
   (2) 012532  005245                        INC     -(R5)
   (2) 012534  000000                        HALT                    ; ADCB INSTRUCTION FAILED
 6789  012536  112710  000377          6$:   MOVB    #377,(R0)       ; LOAD -1
 6790  012542  000261                        SEC                     ; C=1
 6791  012544  105510                        ADCB    (R0)            ; ADD C BIT=1
 6792  012546  004737  017206                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

```
6793                                    ;;*************************************************************************
 (2)                                    ;*TEST: 76      NEW INSTRUCTION IN THIS SECTION IS SBCB
 (3)                                    ;;*************************************************************************
6794
6795  012552                            SBCB1:
 (2)  012552  021527  000076                    CMP     (R5),#76
6796  012556  001404                            BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
6797  012560  012745  000263                    MOV     #263,-(R5)
 (2)  012564  005245                            INC     -(R5)
 (2)  012566  000000                            HALT                            ; TEST IS IN WRONG SEQUENCE
6798  012570  005215                    1$:     INC     (R5)
6799  012572  012701  000444                    MOV     #TEMP2,R1               ; LOAD ADDRESS
6800  012576  112711  000003                    MOVB    #3,(R1)                 ; LOAD LOCATION
6801  012602  000257                            CCC                             ; CLEAR FLAGS
6802  012604  105611                            SBCB    (R1)                    ; SUBTRACT C BIT=0
6803  012606  004737  017062                    JSR     PC,@#$CC0               ; CHECK FOR CC = 0
6804  012612  122711  000003                    CMPB    #3,(R1)                 ; CHECK IT
6805  012616  001404                            BEQ     2$                      ; CONTINUE IF OK
6806  012620  012745  000264                    MOV     #264,-(R5)
 (2)  012624  005245                            INC     -(R5)
 (2)  012626  000000                            HALT                            ; SBCB INSTRUCTION FAILED
6807  012630  000261                    2$:     SEC                             ; C=1
6808  012632  105611                            SBCB    (R1)                    ; SUBTRACT C BIT=1
6809  012634  000261                            SEC                             ; C=1
6810  012636  105611                            SBCB    (R1)                    ;
6811  012640  004737  017062                    JSR     PC,@#$CC0               ; CHECK FOR CC = 0
6812  012644  122711  000001                    CMPB    #1,(R1)                 ; CHECK IT
6813  012650  001404                            BEQ     3$                      ; CONTINUE IF OK
6814  012652  012745  000265                    MOV     #265,-(R5)
 (2)  012656  005245                            INC     -(R5)
 (2)  012660  000000                            HALT                            ; SBCB INSTRUCTION FAILED
6815  012662  000261                    3$:     SEC                             ; C=1
6816  012664  105611                            SBCB    (R1)                    ; SUBTRACT C BIT=1
6817  012666  004737  017164                    JSR     PC,@#$CC4               ; CHECK FOR CC = 4
6818  012672  000261                            SEC                             ; C=1
6819  012674  105611                            SBCB    (R1)                    ; SUBTRACT C BIT = 1
6820  012676  004737  017272                    JSR     PC,@#$CC11              ; CHECK FOR CC = 11
6821  012702  122711  000377                    CMPB    #377,(R1)               ; CHECK IT
6822  012706  001404                            BEQ     4$                      ; CONTINUE IF OK
6823  012710  012745  000266                    MOV     #266,-(R5)
 (2)  012714  005245                            INC     -(R5)
 (2)  012716  000000                            HALT                            ; SBCB INSTRUCTION FAILED
6824  012720  112711  000200            4$:     MOVB    #200,(R1)               ; LOAD R1
6825  012724  000261                            SEC                             ; C=1
6826  012726  105611                            SBCB    (R1)                    ; SUBTRACT C BIT = 1
6827  012730  004737  017122                    JSR     PC,@#$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:
 012734                                         CMP     (R5),#77
 012734  021527  000077
 012740  001404                                 BEQ     1$              ; IF IN WRONG SEQUENCE GO TO HLT
 012742  012745  000267                         MOV     #267,-(R5)
 012746  005245                                 INC     -(R5)
 012750  000000                                 HALT                    ; TEST IS IN A WRONG SEQUENCE
 012752  005215                        1$:     INC     (R5)
 012754  012701  000440                         MOV     #TEMP,R1        ; LOAD ADDRESSES
 012760  012700  000442                         MOV     #TEMP1,R0       ;
 012764  000277                                 SCC
 012766  005010                                 CLR     (R0)            ; CLEAR THE LOCATION
 012770  004737  017164                         JSR     PC,@#$CC4       ; CHECK FOR CC = 4
 012774  005720                                 TST     (R0)+           ; CHECK IT
 012776  004737  017164                         JSR     PC,@#$CC4       ; CHECK FOR CC = 4
 013002  010040                                 MOV     R0,-(R0)
 013004  012730  177777                         MOV     #177777,@(R0)+
 013010  017011  177776                         MOV     @-2(R0),(R1)    ; LOAD THE LOCATION
 013014  004737  017252                         JSR     PC,@#$CC10      ; CHECK FOR CC = 10
 013020  005711                                 TST     (R1)            ; CHECK IT
 013022  004737  017252                         JSR     PC,@#$CC10      ; CHECK FOR CC = 10
```

```
                                ;;*******************************************************************************
                                ;*TEST: 100     NEW INSTRUCTIONS USED IN THIS SECTION ARE CMP, BIS
                                ;;*******************************************************************************
                        CMP1:
                                CMP     (R5),#100
                                BNE     ECMP1                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE EST
                        1$:     INC     (R5)
                                MOV     #TEMP1,R2               ; LOAD ADDRESS
                                MOV     #TEMP,R0                ; PLACE THE ADDRESS OF TEMP IN R0
                                MOV     #177777,(R0)+           ; PLACE #177777 IN LOCATION TEMP AND INC. R0 BY 2
                                BIS     -(R0),(R2)              ; LOAD LOCATION
                                JSR     PC,@#$CC10              ; CHECK FOR CC = 10
                                CMP     (R2)+,#177777           ; CHECK COMPARE
                                BEQ     2$                      ; CONTINUE IF OK
                                MOV     #270,-(R5)
                                INC     -(R5)
                                HALT                            ; CMP OR BIS INSTRUCTION FAILED
                        2$:     CMP     R2,#TEMP1+2             ; CHECK R2 TO CONTAIN ADDRESS OF TEMP1+2
                                BEQ     3$
                                MOV     #271,-(R5)
                                INC     -(R5)
                                HALT                            ; NO AUTO INCREMENT
                        3$:     CMP     #77,-(R2)               ; CHECK IT AGAIN
                                JSR     PC,@#$CC1               ; CHECK FOR CC = 1
                                CMP     #77777,(R2)+
                                JSR     PC,@#$CC13              ; CHECK FOR CC = 13
                                CMP     -(R2),#77777            ; ONCE MORE
                                JSR     PC,@#$CC10              ; CHECK FOR CC = 10
                                MOV     #52525,TEMP2            ; SET EVERY OTHER BIT IN TEMP2
                                MOV     #TEMP2,TEMP1            ; PLACE THE ADDRESS OF TEMP2 IN LOCATION TEMP1
                                MOV     #ADR,R4
                                MOV     #ADR1,(R4)              ; PLACE THE ADDRESS OF ADR1 IN ADR POINTED BY R4
                                MOV     #125252,@(R4)+          ; PLACE THE #125252 IN LOCATION ADR1
                                BIS     @-2(R4),@(R2)+          ; SET EVERY OTHER BIT AT LOCATION TEMP2
                                                                ; AND INCREMENT R2 BY 2
                                MOV     R2,R0                   ; PLACE ADDRESS OF TEMP2 IN R0
                                CMP     @-(R0),#177777          ; TEMP2 SHOULD CONTAIN ALL 1'S
                                BEQ     4$
                                MOV     #272,-(R5)
                                INC     -(R5)
                                HALT                            ; CMP OR BIS INSTUCTIONS FAILED IN MODES OTHER THAN 0
                        4$:     CMP     R2,#TEMP1+2             ; R2 SHOULD CONTAIN THE ADDRESS FOR TEMP2
                                                                ; I.E. TEMP1+2
                                BEQ     5$
                                MOV     #273,-(R5)
                                INC     -(R5)
                                HALT                            ; MODE 5 IS FAILING
                        5$:     CLR     -(R0)                   ; PLACE A 0 IN LOCATION TEMP
                                MOV     R0,TEMP2                ; PLACE ADDRESS OF TEMP IN LOCATION TEMP2
                                CMP     (R0)+,(R0)+             ; BUMP R0 BY 4
                                BIS     @-(R0),@2(R0)           ; PLACE THE CONTENTS OF LOCATION TEMP2 AT TEMP
                                CMP     #TEMP,TEMP              ; LOCATION TEMP SHOULD CONTAIN ITS OWN ADDRESS
                                BEQ     BIC1
                        ECMP1:
                                MOV     #274,-(R5)
```

```
  (2) 013266 005245                    INC     -(R5)
  (2) 013270 000000                    HALT                    ; CMP OR BIS INSTRUCTIONS FAILED OR WRONG
6907                                                           ; SEQUENCE COUNTER
6908
6909
6910
6911                                ;;*************************************************************************
  (2)                               ;*TEST: 101     NEW INSTRUCTIONS USED IN THIS SECTION ARE BIC, BIT
  (3)                               ;;*************************************************************************
6912
6913 013272                         BIC1:
  (2) 013272 021527 000101                CMP     (R5),#101
6914 013276 001122                        BNE     EBIC1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
6915 013300 005215                        INC     (R5)
6916 013302 012703 000440                 MOV     #TEMP,R3        ; LOAD ADDRESS
6917 013306 012713 177777                 MOV     #177777,(R3)    ; LOAD LOCATION
6918 013312 012704 000430                 MOV     #ADR,R4         ; PLACE THE ADDRESS OF ADR IN R4
6919 013316 012714 000432                 MOV     #ADR1,(R4)      ; PLACE THE ADDRESS OF ADR1 IN ADR
6920 013322 011334                        MOV     (R3),@(R4)+     ; LOAD LOCATION ADR1 WITH #177777
6921 013324 012700 000442                 MOV     #TEMP1,R0       ; PLACE THE ADDRESS OF TEMP1 IN R0
6922 013330 012710 125252                 MOV     #125252,(R0)    ; SET EVERY OTHER BIT AT LOCATION TEMP1
6923 013334 000277                        SCC
6924 013336 042013                        BIC     (R0)+,(R3)      ; CLEAR EVERY OTHER BIT
6925 013340 004737 017102                 JSR     PC,@#$CC1       ; CHECK FOR CC = 1
6926 013344 034013                        BIT     -(R0),(R3)      ; CHECK IT
6927 013346 001404                        BEQ     1$              ; CONTINUE IF OK
6928 013350 012745 000275                 MOV     #275,-(R5)
  (2) 013354 005245                       INC     -(R5)
  (2) 013356 000000                       HALT                    ; BIC OR BIT INSTRUCTION FAILED
6929 013360 032713 052525         1$:     BIT     #52525,(R3)     ; CHECK IT
6930 013364 004737 017102                 JSR     PC,@#$CC1       ; CHECK FOR CC = 1
6931 013370 056013 000000                 BIS     0(R0),(R3)      ; SET THE BITS THAT WERE CLEARED
6932 013374 100404                        BMI     2$              ; CONTINUE IF OK
6933 013376 012745 000276                 MOV     #276,-(R5)
  (2) 013402 005245                       INC     -(R5)
  (2) 013404 000000                       HALT                    ; BIT OR BIS INSTRUCTION FAILED
6934 013406 012720 077777         2$:     MOV     #77777,(R0)+    ; SET ALL THE BITS AT LOCATION TEMP1 EXCEPT SIGN BIT
6935 013412 010002                        MOV     R0,R2
6936 013414 046213 177776                 BIC     -2(R2),(R3)     ; TRY CLEARING THE OTHER BITS
6937 013420 004737 017272                 JSR     PC,@#$CC11      ; CHECK FOR CC = 11
6938 013424 020027 000444                 CMP     R0,#TEMP1+2     ; R0 SHOULD CONTAIN THE ADDRESS OF TEMP1+2
6939 013430 001404                        BEQ     3$
6940 013432 012745 000277                 MOV     #277,-(R5)
  (2) 013436 005245                       INC     -(R5)
  (2) 013440 000000                       HALT
6941 013442 010020                3$:     MOV     R0,(R0)+        ; PLACE THE ADDRES OF LOCATION TEMP2 IN TEMP2
6942 013444 000263                        SEVC                    ; SET V & C BITS
6943 013446 045000                        BIC     @-(R0),R0       ; CLEAR R0
6944 013450 004737 017206                 JSR     PC,@#$CC5       ; CHECK FOR CC = 5
6945 013454 037413 177776                 BIT     @-2(R4),(R3)    ; CHECK IT
6946 013460 004737 017272                 JSR     PC,@#$CC11      ; CHECK FOR CC = 11
6947 013464 012746 125252                 MOV     #125252,-(SP)   ; SET EVERY OTHER BIT ON THE STACK
6948 013470 017423 177776                 MOV     @-2(R4),(R3)+   ; SET ALL THE BITS AT LOCATION TEMP
6949 013474 046643 000000                 BIC     0(SP),-(R3)     ; CLEAR EVERY OTHER BIT AT LOCATION TEMP
6950 013500 022327 052525                 CMP     (R3)+,#52525    ; TEMP SHOULD CONTAIN # 52525
6951 013504 001404                        BEQ     4$
```

```
 6952  013506  012745  000300                MOV     #300,-(R5)
   (2) 013512  005245                        INC     -(R5)
   (2) 013514  000000                        HALT                    ; BIC FAILED IN MODE 6
 6953  013516  012700  000446        4$:     MOV     #TEMP2+2,R0     ; PLACE THE ADDRESS OF TEMP2+2 IN R0
 6954  013522  010340                        MOV     R3,-(R0)        ; PLACE THE ADDRESS OF TEMP1 IN TEMP2
 6955  013524  014330                        MOV     -(R3),@(R0)+    ; MOVE # 52525 IN LOCATION TEMP1
 6956  013526  000263                        SEVC                    ; SET V & C BITS
 6957  013530  035026                        BIT     @-(R0),(SP)+    ; BIT TEST TEMP1 WITH STACK AND RESTORE STACK POINTER
 6958  013532  004737  017206                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
 6959  013536  020627  000530                CMP     SP,#START       ; MAKE SURE THAT THE SP IS OK
 6960  013542  001404                        BEQ     INC1
 6961  013544                        EBIC1:
   (2) 013544  012745  000301                MOV     #301,-(R5)
   (2) 013550  005245                        INC     -(R5)
   (2) 013552  000000                        HALT                    ; STACK POINTER FOULED UP OR SEQUENCE ERROR
```

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

013554                                  INC1:
013554  021527  000102                          CMP     (R5),#102
013560  001404                                  BEQ     2$                      ; IF IN WRONG SEQUENCE GO TO HLT BELOW
013562  012745  000302                          MOV     #302,-(R5)
013566  005245                                  INC     -(R5)
013570  000000                                  HALT                            ; PROGRAM IS IN WRONG SEQUENCE
013572  005215                          2$:     INC     (R5)
013574  012704  000442                          MOV     #TEMP1,R4               ; LOAD ADDRESS
013600  012714  077777                          MOV     #77777,(R4)             ; TEMP1 = 77777
013604  000261                                  SEC
013606  005214                                  INC     (R4)                    ; ADD ONES INTO LOCATION
013610  004737  017334                          JSR     PC,@#$CC13              ; CHECK FOR CC = 13
013614  012714  177776                          MOV     #177776,(R4)
013620  012700  000440                          MOV     #TEMP,R0                ; R0 IS POINTING TO LOCATION TEMP
013624  012710  017272                          MOV     #$CC11,(R0)             ; PLACE THE ADDRESS OF SUBROUTINE TO CHECK CC = 11
                                                                                ; IN LOCATION TEMP
013630  005214                                  INC     (R4)                    ;
013632  004730                                  JSR     PC,@(R0)+               ; CHECK FOR CC = 11
013634  005214                                  INC     (R4)                    ;
013636  004737  017206                          JSR     PC,@#$CC5               ; CHECK FOR CC = 5
013642  005214                                  INC     (R4)                    ;
013644  004737  017102                          JSR     PC,@#$CC1               ; CHECK FOR CC = 1
013650  026427  000000  000001                  CMP     0(R4),#1                ; CHECK IT
013656  001404                                  BEQ     4$                      ; CONTINUE IF OK
013660  012745  000303                          MOV     #303,-(R5)
013664  005245                                  INC     -(R5)
013666  000000                                  HALT                            ; INC INSTRUCTION FAILED
013670  000261                          4$:     SEC
013672  005314                                  DEC     (R4)                    ; SUBTRACT ONES FROM LOCATION
013674  004737  017206                          JSR     PC,@#$CC5               ; CHECK FOR CC = 5
013700  005314                                  DEC     (R4)                    ;
013702  004770  177776                          JSR     PC,@-2(R0)              ; CHECK FOR CC = 11
013706  012714  100000                          MOV     #100000,(R4)
013712  005314                                  DEC     (R4)                    ;
013714  004737  017142                          JSR     PC,@#$CC3               ; CHECK FOR CC = 3
013720  005314                                  DEC     (R4)                    ;
013722  004737  017102                          JSR     PC,@#$CC1               ; CHECK FOR CC = 1



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

013726                                  COM1:
013726  021527  000103                          CMP     (R5),#103
013732  001404                                  BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
013734  012745  000304                          MOV     #304,-(R5)
013740  005245                                  INC     -(R5)
013742  000000                                  HALT                            ; TEST IS IN WRONG SEQUENCE
013744  005215                          1$:     INC     (R5)
013746  012703  000442                          MOV     #TEMP1,R3               ; LOAD ADDRESS
```

```
 7006  013752  012713  125252                 MOV     #125252,(R3)    ; LOAD EVERY OTHER BIT
 7007  013756  000277                         SCC
 7008  013760  005163  000000                 COM     0(R3)           ; 1'S COMPLEMENT
 7009  013764  004737  017102                 JSR     PC,@#$CC1       ; CHECK FOR CC = 1
 7010  013770  022713  052525                 CMP     #52525,(R3)     ; CHECK IT
 7011  013774  001404                         BEQ     2$              ; CONTINUE IF OK
 7012  013776  012745  000305                 MOV     #305,-(R5)
   (2) 014002  005245                         INC     -(R5)
   (2) 014004  000000                         HALT                    ; COM INSTRUCTION FAILED
 7013  014006  000277              2$:        SCC
 7014  014010  005123                         COM     (R3)+           ; COMPLEMENT BACK
 7015  014012  004737  017272                 JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 7016  014016  022743  125252                 CMP     #125252,-(R3)   ; CHECK IT
 7017  014022  001404                         BEQ     3$              ; CONTINUE IF OK
 7018  014024  012745  000306                 MOV     #306,-(R5)
   (2) 014030  005245                         INC     -(R5)
   (2) 014032  000000                         HALT                    ; COM INSTRUCTION FAILED
 7019  014034  010300              3$:        MOV     R3,R0           ; R0 IS NOW POINTING TO LOCATION TEMP1
 7020  014036  012710  177777                 MOV     #177777,(R0)
 7021  014042  000277                         SCC
 7022  014044  005110                         COM     (R0)
 7023  014046  004737  017206                 JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

```
7024                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 104     NEW INSTRUCTION IN THIS SECTION IS NEG
 (3)                                    ;;*******************************************************************************
7025
7026  014052                            NEG1:
 (2)  014052  021527  000104                    CMP     (R5),#104
7027  014056  001033                            BNE     ENEG1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
7028  014060  005215                    1$:     INC     (R5)
7029  014062  012704  000442                    MOV     #TEMP1,R4       ; LOAD ADDRESS
7030  014066  012724  000001                    MOV     #1,(R4)+                ; LOAD THE LOCATION
7031  014072  010402                            MOV     R4,R2
7032  014074  012762  100000  000000            MOV     #100000,0(R2)
7033  014102  005444                            NEG     -(R4)           ; 2'S COMPLEMENT
7034  014104  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
7035  014110  022724  177777                    CMP     #177777,(R4)+   ; CHECK IT
7036  014114  001404                            BEQ     2$              ; CONTINUE IF OK
7037  014116  012745  000307                    MOV     #307,-(R5)
 (2)  014122  005245                            INC     -(R5)
 (2)  014124  000000                            HALT                    ; NEG INSTRUCTION FAILED
7038  014126  016444  000000            2$:     MOV     0(R4),-(R4)     ; TEMP1 CONTAINS THE LARGEST NEGATIVE NUMBER
7039  014132  005414                            NEG     (R4)            ; 2'S COMPLEMENT
7040  014134  004737  017334                    JSR     PC,@#$CC13      ; CHECK FOR CC = 13
7041  014140  026214  000000                    CMP     0(R2),(R4)      ; CHECK IT
7042  014144  001404                            BEQ     ROL1            ; CONTINUE IF OK
7043  014146                            ENEG1:
 (2)  014146  012745  000310                    MOV     #310,-(R5)
 (2)  014152  005245                            INC     -(R5)
 (2)  014154  000000                            HALT                    ; WRONG RESULT IN TEMP2 OR WRONG SEQUENCE
7044
7045
7046
7047                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 105     NEW INSTRUCTION IN THIS SECTION IS ROL
 (3)                                    ;;*******************************************************************************
7048
7049  014156                            ROL1:
 (2)  014156  021527  000105                    CMP     (R5),#105
7050  014162  001032                            BNE     EROL1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
7051  014164  005215                            INC     (R5)
7052  014166  012701  000444                    MOV     #TEMP2,R1       ; LOAD ADDRESS
7053  014172  012711  020000                    MOV     #20000,(R1)     ; LOAD LOCATION
7054  014176  000257                            CCC                     ; CLEAR FLAGS
7055  014200  006121                            ROL     (R1)+           ; SHIFT
7056  014202  006141                            ROL     -(R1)           ;
7057  014204  004737  017314                    JSR     PC,@#$CC12      ; CHECK FOR CC = 12
7058  014210  022711  100000                    CMP     #100000,(R1)    ; CHECK IT
7059  014214  001404                            BEQ     1$              ; CONTINUE IF OK
7060  014216  012745  000311                    MOV     #311,-(R5)
 (2)  014222  005245                            INC     -(R5)
 (2)  014224  000000                            HALT                    ; ROL INSTRUCTION FAILED
7061  014226  006161  000000            1$:     ROL     0(R1)           ; SHIFT
7062  014232  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
7063  014236  010102                            MOV     R1,R2           ; R2 IS NOW POINTING TO LOCATION TEMP2
7064  014240  006112                            ROL     (R2)            ; SHIFT
7065  014242  022711  000001                    CMP     #1,(R1)         ; CHECK IT
7066  014246  001404                            BEQ     ROR1            ; CONTINUE IF OK
```

```
 7067  014250                        EROL1:
  (2)  014250  012745  000312                MOV     #312,-(R5)
  (2)  014254  005245                        INC     -(R5)
  (2)  014256  000000                        HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE
```

```
7068                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 106     NEW INSTRUCTION IN THIS SECTION IS ROR
 (3)                                    ;;*******************************************************************************
7069
7070  014260                            ROR1:
 (2)  014260  021527  000106                    CMP     (R5),#106
7071  014264  001030                            BNE     EROR1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
7072  014266  005215                            INC     (R5)
7073  014270  012702  000444                    MOV     #TEMP2,R2       ; LOAD ADDRESS
7074  014274  012712  000004                    MOV     #4,(R2)         ; LOAD LOCATION
7075  014300  000257                            CCC                     ; CLEAR FLAGS
7076  014302  006012                            ROR     (R2)            ; SHIFT
7077  014304  006012                            ROR     (R2)            ;
7078  014306  022712  000001                    CMP     #1,(R2)         ; CHECK IT
7079  014312  001404                            BEQ     1$              ; CONTINUE IF OK
7080  014314  012745  000313                    MOV     #313,-(R5)
 (2)  014320  005245                            INC     -(R5)
 (2)  014322  000000                            HALT                    ; ROR INSTRUCTION FAILED
7081  014324  006012                    1$:     ROR     (R2)            ; SHIFT
7082  014326  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
7083  014332  006012                            ROR     (R2)            ; SHIFT
7084  014334  004737  017314                    JSR     PC,@#$CC12      ; CHECK FOR CC = 12
7085  014340  022712  100000                    CMP     #100000,(R2)    ; CHECK IT
7086  014344  001404                            BEQ     ASL1            ; CONTINUE IF OK
7087  014346                            EROR1:
 (2)  014346  012745  000314                    MOV     #314,-(R5)
 (2)  014352  005245                            INC     -(R5)
 (2)  014354  000000                            HALT                    ; WRONG RESULT AT TEMP2 OR WRONG SEQUENCE
7088
7089
7090
7091                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 107     NEW INSTRUCTION IN THIS SECTION IS ASL
 (3)                                    ;;*******************************************************************************
7092
7093  014356                            ASL1:
 (2)  014356  021527  000107                    CMP     (R5),#107
7094  014362  001404                            BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
7095  014364  012745  000315                    MOV     #315,-(R5)
 (2)  014370  005245                            INC     -(R5)
 (2)  014372  000000                            HALT                    ; PROGRAM IS IN WRONG SEQUENCE
7096  014374  005215                    2$:     INC     (R5)
7097  014376  012703  000444                    MOV     #TEMP2,R3       ; LOAD ADDRESS
7098  014402  012713  020000                    MOV     #20000,(R3)     ; LOAD LOCATION
7099  014406  000257                            CCC                     ; CLEAR FLAGS
7100  014410  006313                            ASL     (R3)            ; SHIFT
7101  014412  006313                            ASL     (R3)            ;
7102  014414  004737  017314                    JSR     PC,@#$CC12      ; CHECK FOR CC = 12
7103  014420  022713  100000                    CMP     #100000,(R3)    ; CHECK IT
7104  014424  001404                            BEQ     4$              ; CONTINUE IF OK
7105  014426  012745  000316                    MOV     #316,-(R5)
 (2)  014432  005245                            INC     -(R5)
 (2)  014434  000000                            HALT                    ; ASL INSTRUCTION FAILED
7106  014436  006313                    4$:     ASL     (R3)            ; SHIFT
7107  014440  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
7108  014444  006313                            ASL     (R3)            ; SHIFT
```

N06


```
  7109  014446  004737  017164                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
```

```
7110                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 110     NEW INSTRUCTION IN THIS SECTION IS ASR
 (3)                                    ;;*******************************************************************************
7111
7112  014452                           ASR1:
 (2)  014452  021527  000110                    CMP     (R5),#110
7113  014456  001040                           BNE     EASR1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
7114  014460  005215                    1$:     INC     (R5)
7115  014462  012704  000444                    MOV     #TEMP2,R4       ; LOAD ADDRESSES
7116  014466  012703  000440                    MOV     #TEMP,R3        ;
7117  014472  012714  000004                    MOV     #4,(R4)         ; LOAD LOCATION
7118  014476  000257                           CCC                     ; CLEAR FLAGS
7119  014500  006214                           ASR     (R4)            ; SHIFT
7120  014502  006214                           ASR     (R4)            ;
7121  014504  022714  000001                    CMP     #1,(R4)         ; CHECK IT
7122  014510  001404                           BEQ     2$              ; CONTINUE IF OK
7123  014512  012745  000317                    MOV     #317,-(R5)
 (2)  014516  005245                           INC     -(R5)
 (2)  014520  000000                           HALT                    ; ASR INSTRUCTION FAILED
7124  014522  006214                    2$:     ASR     (R4)            ; SHIFT
7125  014524  004737  017230                    JSR     PC,@#$CC7       ; CHECK FOR CC = 7
7126  014530  006214                           ASR     (R4)            ; SHIFT
7127  014532  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
7128  014536  012713  100002                    MOV     #100002,(R3)    ; LOAD LOCATION
7129  014542  006213                           ASR     (R3)            ; SHIFT
7130  014544  006213                           ASR     (R3)            ;
7131  014546  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11
7132  014552  022713  160000                    CMP     #160000,(R3)    ; CHECK IT
7133  014556  001404                           BEQ     ADC1            ; CONTINUE IF OK
7134  014560                           EASR1:
 (2)  014560  012745  000320                    MOV     #320,-(R5)
 (2)  014564  005245                           INC     -(R5)
 (2)  014566  000000                           HALT                    ; WRONG RESULT IN TEMP OR WRONG SEQUENCE
7135
7136
7137
7138                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 111     NEW INSTRUCTION IN THIS SECTION IS ADC
 (3)                                    ;;*******************************************************************************
7139
7140  014570                           ADC1:
 (2)  014570  021527  000111                    CMP     (R5),#111
7141  014574  001404                           BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
7142  014576  012745  000321                    MOV     #321,-(R5)
 (2)  014602  005245                           INC     -(R5)
 (2)  014604  000000                           HALT                    ; PROGRAM IS IN WRONG SEQUENCE
7143  014606  005215                    2$:     INC     (R5)
7144  014610  012700  000440                    MOV     #TEMP,R0        ; LOAD ADDRESS
7145  014614  005010                           CLR     (R0)            ; CLEAR THE LOCATION
7146  014616  000257                           CCC                     ; CLEAR FLAGS
7147  014620  005510                           ADC     (R0)            ; ADD C BIT = 0
7148  014622  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
7149  014626  000261                           SEC                     ; C=1
7150  014630  005510                           ADC     (R0)            ; ADD C BIT=1
7151  014632  000261                           SEC                     ; C=1
7152  014634  005510                           ADC     (R0)            ; AGAIN
```

C07


```
 7153  014636  004737  017062                JSR     PC,@#$CC0       ; CHECK FOR CC = 0
 7154  014642  022710  000002                CMP     #2,(R0)         ; CHECK IT
 7155  014646  001404                        BEQ     4$              ; CONTINUE IF OK
 7156  014650  012745  000322                MOV     #322,-(R5)
   (2) 014654  005245                        INC     -(R5)
   (2) 014656  000000                        HALT                    ; ADC INSTRUCTION FAILED
 7157  014660  012710  077777          4$:   MOV     #77777,(R0)     ; LOAD LARGEST POSITVE NUMBER
 7158  014664  000261                        SEC                     ; C=1
 7159  014666  005510                        ADC     (R0)            ; ADD C BIT=1
 7160  014670  004737  017314                JSR     PC,@#$CC12      ; CHECK FOR CC = 12
 7161  014674  022710  100000                CMP     #100000,(R0)    ; CHECK IT
 7162  014700  001404                        BEQ     6$              ; CONTINUE IF OK
 7163  014702  012745  000323                MOV     #323,-(R5)
   (2) 014706  005245                        INC     -(R5)
   (2) 014710  000000                        HALT                    ; ADC INSTRUCTION AILED
 7164  014712  012710  177777          6$:   MOV     #-1,(R0)        ; LOAD -1
 7165  014716  000261                        SEC                     ; C=1
 7166  014720  005510                        ADC     (R0)            ; ADD C BIT=1
 7167  014722  004737  017206                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
```

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

014726                          SBC1:
014726  021527  000112                  CMP     (R5),#112
014732  001404                          BEQ     1$                      ; IF IN WRONG SEQUENCE GO TO HLT
014734  012745  000324                  MOV     #324,-(R5)
014740  005245                          INC     -(R5)
014742  000000                          HALT                            ; TEST IS IN WRONG SEQUENCE
014744  005215                  1$:     INC     (R5)
014746  012701  000440                  MOV     #TEMP,R1                ; LOAD ADDRESS
014752  012711  000003                  MOV     #3,(R1)                 ; LOAD LOCATION
014756  000257                          CCC                             ; CLEAR FLAGS
014760  005611                          SBC     (R1)                    ; SUBTRACT C BIT=0
014762  004737  017062                  JSR     PC,@#$CCO               ; CHECK FOR CC = 0
014766  022711  000003                  CMP     #3,(R1)                 ; CHECK IT
014772  001404                          BEQ     2$                      ; CONTINUE IF OK
014774  012745  000325                  MOV     #325,-(R5)
015000  005245                          INC     -(R5)
015002  000000                          HALT                            ; SBC INSTRUCTION FAILED
015004  000261                  2$:     SEC                             ; C=1
015006  005611                          SBC     (R1)                    ; SUBTRACT C BIT=1
015010  000261                          SEC                             ; C=1
015012  005611                          SBC     (R1)                    ;
015014  004737  017062                  JSR     PC,@#$CCO               ; CHECK FOR CC = 0
015020  022711  000001                  CMP     #1,(R1)                 ; CHECK IT
015024  001404                          BEQ     3$                      ;
015026  012745  000326                  MOV     #326,-(R5)
015032  005245                          INC     -(R5)
015034  000000                          HALT                            ; SBC INSTRUCTION FAILED
015036  000261                  3$:     SEC                             ; C=1
015040  005611                          SBC     (R1)                    ; SUBTRACT C BIT=1
015042  004737  017164                  JSR     PC,@#$CC4               ; CHECK FOR CC = 4
015046  000261                          SEC                             ; C=1
015050  005611                          SBC     (R1)                    ; SUBTRACT C BIT = 1
015052  004737  017272                  JSR     PC,@#$CC11              ; CHECK FOR CC = 11
015056  022711  177777                  CMP     #-1,(R1)                ; CHECK IT
015062  001404                          BEQ     4$                      ; CONTINUE IF OK
015064  012745  000327                  MOV     #327,-(R5)
015070  005245                          INC     -(R5)
015072  000000                          HALT                            ; SBC INSTRUCTION FAILED
015074  012711  100000          4$:     MOV     #100000,(R1)            ; LOAD R1
015100  000261                          SEC                             ; C=1
015102  005611                          SBC     (R1)                    ; SUBTRACT C BIT = 1
015104  004737  017122                  JSR     PC,@#$CC2               ; CHECK FOR CC = 2



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

015110                          SXT1:
015110  021527  000113                  CMP     (R5),#113
015114  001026                          BNE     ESXT1                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
```

E07


```
 7210  015116  005215                1$:     INC     (R5)
 7211  015120  012702  000442                MOV     #TEMP1,R2       ; LOAD ADDRESS
 7212  015124  005012                        CLR     (R2)            ; CLEAR LOCATIONS
 7213  015126  000277                        SCC
 7214  015130  000254                        CLNZ
 7215  015132  006712                        SXT     (R2)            ; SIGN EXTEND
 7216  015134  004737  017206                JSR     PC,@#$CC5       ; CHECK FOR CC = 5
 7217  015140  005712                        TST     (R2)            ; LOCATION SHOULD STILL BE 0
 7218  015142  001404                        BEQ     2$              ; CONTINUE IF OK
 7219  015144  012745  000330                MOV     #330,-(R5)
   (2) 015150  005245                        INC     -(R5)
   (2) 015152  000000                        HALT                    ; SXT INSTRUCTION FAILED
 7220  015154  000273                2$:     SENVC                   ; SET N, V & C BITS
 7221  015156  006712                        SXT     (R2)            ; SIGN EXTEND
 7222  015160  004737  017272                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 7223  015164  022712  177777                CMP     #-1,(R2)        ; LOCATION SHOULD NEW HAVE -1
 7224  015170  001404                        BEQ     SWAB1           ; CONTINUE IF OK
 7225  015172                        ESXT1:
   (2) 015172  012745  000331                MOV     #331,-(R5)
   (2) 015176  005245                        INC     -(R5)
   (2) 015200  000000                        HALT                    ; WRONG RESULT IN TEMP1 OR WRONG SEQUENCE
```

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

015202                                  SWAB1:
015202  021527  000114                          CMP     (R5),#114
015206  001034                                  BNE     ESWAB1                  ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
015210  005215                                  INC     (R5)
015212  012703  000444                          MOV     #TEMP2,R3               ; LOAD ADDRESS
015216  012713  125125                          MOV     #125125,(R3)            ; LOAD BIT PATTERN INTO LOCATION
015222  000277                                  SCC
015224  000250                                  CLN
015226  000313                                  SWAB    (R3)                    ; SWAP BYTES OF LOCATIONS
015230  004737  017252                          JSR     PC,@#$CC10              ; CHECK FOR CC = 10
015234  022713  052652                          CMP     #52652,(R3)             ; CHECK IT
015240  001404                                  BEQ     1$                      ; CONTINUE IF OK
015242  012745  000332                          MOV     #332,-(R5)
015246  005245                                  INC     -(R5)
015250  000000                                  HALT                            ; SWAB INSTRUCTION FAILED
015252  012713  000377                  1$:     MOV     #377,(R3)
015256  000277                                  SCC
015260  000244                                  CLZ
015262  000363  000000                          SWAB    0(R3)
015266  004737  017164                          JSR     PC,@#$CC4               ; CHECK FOR CC = 4
015272  022713  177400                          CMP     #177400,(R3)
015276  001404                                  BEQ     XOR1
015300                                  ESWAB1:
015300  012745  000333                          MOV     #333,-(R5)
015304  005245                                  INC     -(R5)
015306  000000                                  HALT                            ; WRONG RESULT IN: TEMP2 OR WRONG SEQUENCE



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

015310                                  XOR1:
015310  021527  000115                          CMP     (R5),#115
015314  001041                                  BNE     EXOR1                   ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
015316  005215                                  INC     (R5)
015320  012704  177777                          MOV     #-1,R4                  ; LOAD LOCATIONS
015324  012767  177777  163110                  MOV     #-1,TEMP1               ;
015332  000277                                  SCC
015334  074467  163102                          XOR     R4,TEMP1                ; SHOULD PRODUCE 0'S IN TEMP1
015340  004737  017206                          JSR     PC,@#$CC5               ; CHECK FOR CC = 5
015344  012767  077777  163070                  MOV     #77777,TEMP1
015352  012700  000442                          MOV     #TEMP1,R0               ; PLACE THE ADDRESS OF TEMP1 IN R0
015356  000263                                  SEVC                            ; SET V & C BITS
015360  000244                                  CLZ
015362  074410                                  XOR     R4,(R0)
015364  004737  017272                          JSR     PC,@#$CC11              ; CHECK FOR CC = 11
015370  012701  125252                          MOV     #125252,R1              ; LOAD LOCATIONS
015374  012720  052525                          MOV     #52525,(R0)+            ;
015400  000277                                  SCC
015402  074140                                  XOR     R1,-(R0)                ; SHOULD PRODUCE ALL 1'S IN TEMP1
```

```
 7271  015404  004737  017272                JSR     PC,@#$CC11      ; CHECK FOR CC = 11
 7272  015410  022737  177777  000442        CMP     #-1,@#TEMP1     ; CHECK IT
 7273  015416  001404                        BEQ     ADD1            ; CONTINUE IF OK
 7274  015420                          EXOR1:
   (2) 015420  012745  000334                MOV     #334,-(R5)
   (2) 015424  005245                        INC     -(R5)
   (2) 015426  000000                        HALT                    ; WRONG RESULT IN TEMP1 OR WRONG SEQUENCE
```

```
7275                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 116     NEW INSTRUCTION IN THIS SECTION IS ADD
 (3)                                    ;;*******************************************************************************
7276
7277  015430                            ADD1:
 (2)  015430  021527  000116                    CMP     (R5),#116
7278  015434  001133                           BNE     EADD1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
7279  015436  005215                           INC     (R5)
7280  015440  012700  000444                   MOV     #TEMP2,R0       ; LOAD ADDRESSES
7281  015444  012701  000440                   MOV     #TEMP,R1        ;
7282  015450  012767  021421  162766           MOV     #21421,TEMP2    ; LOAD LOCATIONS
7283  015456  011011                           MOV     (R0),(R1)       ;
7284  015460  061011                           ADD     (R0),(R1)       ; ADD
7285  015462  004737  017062                   JSR     PC,@#$CCO       ; CHECK FOR CC = 0
7286  015466  022767  043042  162744           CMP     #43042,TEMP     ; CHECK IT
7287  015474  001404                           BEQ     1$              ; CONTINUE IF OK
7288  015476  012745  000335                   MOV     #335,-(R5)
 (2)  015502  005245                           INC     -(R5)
 (2)  015504  000000                           HALT                    ; ADD INSTRUCTION FAILED
7289  015506  005010                    1$:    CLR     (R0)            ; CLEAR LOCATION TEMP2
7290  015510  060020                           ADD     R0,(R0)+        ; PLACE THE ADDRESS OF TEMP2 IN LOCATION TEMP2
7291  015512  024027  000444                   CMP     -(R0),#TEMP2    ; CHECK IT
7292  015516  001404                           BEQ     2$
7293  015520  012745  000336                   MOV     #336,-(R5)
 (2)  015524  005245                           INC     -(R5)
 (2)  015526  000000                           HALT                    ; ADD INSTRUCTION FAILED IN MODE 2
7294  015530  012767  156357  162706    2$:    MOV     #-21421,TEMP2   ; LOAD LOCATIONS
7295  015536  012011                           MOV     (R0)+,(R1)      ;
7296  015540  064011                           ADD     -(R0),(R1)      ; ADD
7297  015542  004737  017272                   JSR     PC,@#$CC11      ; CHECK FOR CC = 11
7298  015546  022767  134736  162664           CMP     #-43042,TEMP    ; CHECK IT
7299  015554  001404                           BEQ     3$              ;CONTINUE IF OK
7300  015556  012745  000337                   MOV     #337,-(R5)
 (2)  015562  005245                           INC     -(R5)
 (2)  015564  000000                           HALT                    ; ADD INSTRUCTION FAILED
7301  015566  012767  100000  162650    3$:    MOV     #100000,TEMP2   ; LOAD LOCATIONS
7302  015574  011061  000000                   MOV     (R0),0(R1)      ;
7303  015600  066011  000000                   ADD     0(R0),(R1)      ; ADD SHOULD RESULT AS 0'S
7304  015604  004737  017230                   JSR     PC,@#$CC7       ; CHECK FOR CC=7
7305  015610  012767  021421  162624           MOV     #21421,TEMP1    ; LOAD LOCATION TEMP1
7306  015616  012760  000442  000000           MOV     #TEMP1,0(R0)    ; PLACE THE ADDRESS OF LOCATION TEMP1 IN TEMP2
7307  015624  012711  156357                   MOV     #-21421,(R1)    ; LOAD LOCATION TEMP
7308  015630  010004                           MOV     R0,R4           ; MAKE R4 POINT TO LOCATION TEMP2
7309  015632  067411  000000                   ADD     @0(R4),(R1)     ; ADD SHOULD RESULT AS 0'S
7310  015636  004737  017206                   JSR     PC,@#$CC5       ; CHECK FOR CC=5
7311  015642  005430                           NEG     @(R0)+          ; NEGATE THE CONTENTS OF TEMP1
7312  015644  012746  021421                   MOV     #21421,-(SP)    ; PLACE # 21421 ON THE STACK
7313  015650  065066  000000                   ADD     @-(R0),0(SP)    ; ADD, SHOULD=0'S
7314  015654  004737  017206                   JSR     PC,@#$CC5       ; CHECK FOR CC=5
7315  015660  005726                           TST     (SP)+           ; CHECK THE STACK TO CONTAIN 0, ALSO
7316                                                                   ; RESTORE THE STACK POINTER
7317  015662  001404                           BEQ     4$
7318  015664  012745  000340                   MOV     #340,-(R5)
 (2)  015670  005245                           INC     -(R5)
 (2)  015672  000000                           HALT                    ; ADD INSTRUCTION FAILED IN MODE 5
7319  015674  012767  137777  162542    4$:    MOV     #137777,TEMP2
```

```
 7320  015702  062767  137777  162534          ADD     #137777,TEMP2
 7321  015710  004737  017142                  JSR     PC,@#$CC3       ; CHECK CC=3
 7322  015714  022767  077776  162522          CMP     #77776,TEMP2
 7323  015722  001404                          BEQ     SUB1
 7324  015724                          EADD1:
   (2) 015724  012745  000341                  MOV     #341,-(R5)
   (2) 015730  005245                          INC     -(R5)
   (2) 015732  000000                          HALT                    ; WRONG RESULT AT TEMP OR WRONG SEQUENCE
 7325
 7326
 7327
 7328                                  ;;*******************************************************************************
   (2)                                 ;*TEST: 117     NEW INSTRUCTION IN THIS SECTION IS SUB
   (3)                                 ;;*******************************************************************************
 7329
 7330  015734                          SUB1:
   (2) 015734  021527  000117                  CMP     (R5),#117
 7331  015740  001100                          BNE     ESUB1           ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
 7332  015742  005215                          INC     (R5)
 7333  015744  012702  000440                  MOV     #TEMP,R2        ; LOAD ADDRESSES
 7334  015750  012703  000442                  MOV     #TEMP1,R3       ;
 7335  015754  012767  021421  162456          MOV     #21421,TEMP     ; LOAD LOCATIONS
 7336  015762  012767  156357  162452          MOV     #-21421,TEMP1   ;
 7337  015770  161213                          SUB     (R2),(R3)       ; RESULT SHOULD=-43042
 7338  015772  004737  017252                  JSR     PC,@#$CC10      ; CHECK FOR CC = 10
 7339  015776  022767  134736  162436          CMP     #-43042,TEMP1   ; CHECK IT
 7340  016004  001404                          BEQ     1$              ;CONTINUE IF OK
 7341  016006  012745  000342                  MOV     #342,-(R5)
   (2) 016012  005245                          INC     -(R5)
   (2) 016014  000000                          HALT                    ; SUB INSTRUCTION FAILED
 7342  016016  012767  021421  162416  1$:     MOV     #21421,TEMP1    ; LOAD LOCATION
 7343  016024  161213                          SUB     (R2),(R3)       ; RESULT SHOULD=0
 7344  016026  001404                          BEQ     2$
 7345  016030  012745  000343                  MOV     #343,-(R5)
   (2) 016034  005245                          INC     -(R5)
   (2) 016036  000000                          HALT                    ; SUB INSTRUCTION FAILED
 7346  016040  012767  177777  162374  2$:     MOV     #-1,TEMP1       ; LOAD LOCATIONS
 7347  016046  012767  077777  162364          MOV     #77777,TEMP     ; LOAD LOCATIONS
 7348  016054  161312                          SUB     (R3),(R2)       ; RESULT SHOULD GIVE 100000 AND OVERFLOW
 7349  016056  004737  017334                  JSR     PC,@#$CC13      ; CHECK FOR CC = 13
 7350  016062  022767  100000  162350          CMP     #100000,TEMP    ; CHECK IT
 7351  016070  001404                          BEQ     3$              ; CONTINUE IF OK
 7352  016072  012745  000344                  MOV     #344,-(R5)
   (2) 016076  005245                          INC     -(R5)
   (2) 016100  000000                          HALT                    ; SUB INSTRUCTION FAILED
 7353  016102  012712  177777          3$:     MOV     #-1,(R2)
 7354  016106  161312                          SUB     (R3),(R2)
 7355  016110  004737  017164                  JSR     PC,@#$CC4       ; CHECK FOR CC = 4
 7356  016114  012767  077777  162316          MOV     #77777,TEMP
 7357  016122  162767  077777  162310          SUB     #77777,TEMP
 7358  016130  004737  017164                  JSR     PC,@#$CC4       ; CHECK FOR CC=4
 7359  016134  005767  162300                  TST     TEMP
 7360  016140  001404                          BEQ     SOB             ; TEMP SHOULD BE =0
 7361  016142                          ESUB1:
   (2) 016142  012745  000345                  MOV     #345,-(R5)
   (2) 016146  005245                          INC     -(R5)
```

```
DVKAAA  MACY11 27(732)  25-AUG-76  13:25  PAGE 54-84
DVKAAA.P11      T117    NEW INSTRUCTION IN THIS SECTION IS SUB                    *** SEQ 0087

   (2)  016150  000000                        HALT                ; SUB INSTRUCTION FAILED OR SEQUENCE ERROR
```

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

016152                           SOB:
016152  021527  000120                   CMP     (R5),#120
016156  001042                           BNE     ESOB                    ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
016160  005215                           INC     (R5)
016162  012700  000012                   MOV     #10.,R0                 ; LOAD REGISTERS
016166  005001                           CLR     R1                      ;
016170  005201                   1$:     INC     R1                      ; KEEP COUNT
016172  020127  000012                   CMP     R1,#10.
016176  003404                           BLE     2$
016200  012745  000346                   MOV     #346,-(R5)
016204  005245                           INC     -(R5)
016206  000000                           HALT                            ; SOB INSTRUCTION FAILED
016210  000277                   2$:     SCC
016212  077012                           SOB     R0,1$                   ; SUB. 1 FROM REG. 0, GO BACK TO 1$
016214  004737  017354                   JSR     PC,@#$CC17              ; CHECK FOR CC = 17
016220  005700                           TST     R0                      ; REG. 0 = 0 ?
016222  001404                           BEQ     3$                      ; NO, FAILED
016224  012745  000347                   MOV     #347,-(R5)
016230  005245                           INC     -(R5)
016232  000000                           HALT                            ; SOB INSTRUCTION FAILED
016234  022701  000012           3$:     CMP     #10.,R1                 ; DID IT GO THRU 10 TIMES ?
016240  001404                           BEQ     4$                      ; CONTINUE IF OK
016242  012745  000350                   MOV     #350,-(R5)
016246  005245                           INC     -(R5)
016250  000000                           HALT                            ; SOB INSTRUCTION FAILED
016252  012704  000010           4$:     MOV     #10,R4                  ; PLACE #10 IN R4
016256  077401                   5$:     SOB     R4,5$                   ; STAY HERE UNTILL R4 = 0
016260  005704                           TST     R4
016262  001404                           BEQ     PSWNO                   ; CONTINUE IF OK
016264                           ESOB:
016264  012745  000351                   MOV     #351,-(R5)
016270  005245                           INC     -(R5)
016272  000000                           HALT                            ; SOB FAILED OR WRONG SEQUENCE




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


016274                           PSWNO:
016274  021527  000121                   CMP     (R5),#121
016300  001042                           BNE     EPSWNO                  ; IF IN WRONG SEQUENCE GO TO HLT AT THE END OF THE TEST
016302  005215                           INC     (R5)
016304  012700  000440                   MOV     #TEMP,R0                ; PUT THE ADDRESS OF TEMP IN R0
016310  012701  000442                   MOV     #TEMP1,R1               ; PUT THE ADDRESS OF TEMP1 IN R1
016314  012711  177777                   MOV     #177777,(R1)            ; TEMP1 = 177777
016320  005010                           CLR     (R0)                    ; TEMP  = 0
016322                                   MTPS    (R0)                    ; PSW = 0
016322  106410                           .WORD   106400!..C
016324  004737  017062                   JSR     PC,@#$CC0               ; CHECK FOR CC = 0
```

```
 7405  016330                                    MFPS    (R1)            ; MOVE PSW TO TEMP1
  (1)  016330  106711                            .WORD   106700!..C
 7409  016332  004737  017164                    JSR     PC,@#$CC4       ; CHECK FOR CC = 4
 7410  016336  022711  177400                    CMP     #177400,(R1)    ; CHECK TEMP1 TO MAKE SURE THAT  ONLY
 7411                                                                    ; THE LOWER BYTE WAS AFFECTED BY MFPS
 7412  016342  001404                            BEQ     1$
 7413  016344  012745  000352                    MOV     #352,-(R5)
  (2)  016350  005245                            INC     -(R5)
  (2)  016352  000000                            HALT                    ; MTPS OR MFPS INSTRUCTION FAILED
 7414  016354  005011                    1$:     CLR     (R1)
 7418  016356                                    MTPS    #377            ; SET PSW = 357 SINCE T BIT CAN NOT BE SET BY MTPS
  (1)  016356  106427                            .WORD   106400!..C
 7419  016362  004737  017354                    JSR     PC,@#$CC17      ; CHECK FOR CC = 17
 7420  016366                                    MFPS    TEMP1           ; MOVE PSW TO TEMP1
  (1)  016366  106767                            .WORD   106700!..C
 7424  016372  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC = 11 [C BIT SHOULD NOT BE EFFECTED BY MFP
 7425  016376  022767  000357  162036            CMP     #357,TEMP1
 7426  016404  001404                            BEQ     BTWRD
 7427  016406                            EPSWND:
  (2)  016406  012745  000353                    MOV     #353,-(R5)
  (2)  016412  005245                            INC     -(R5)
  (2)  016414  000000                            HALT                    ; MFPS INSTRUCTION FAILED IN MODE 6
 7428                                                                    ; OR SEQUENCE ERROR
```

```
7429                                    ;;*******************************************************************************
 (2)                                    ;*TEST: 122     BYTE INSTRUCTIONS REQUIRING WORD INST. TO CHECK
 (3)                                    ;;*******************************************************************************
7430
7431  016416                            BTWRD:
 (2)  016416  021527  000122                    CMP     (R5),#122
7432  016422  001124                            BNE     EBTWRD          ; IF IN WRONG SEQUNCE GO TO HALT AT THE END OF THE TEST
7433  016424  005215                            INC     (R5)
7434  016426  005000                            CLR     R0
7435  016430  000277                            SCC
7436  016432  112700  000200                   MOVB    #200,R0         ; SET THE HIGHEST BIT OF THE
7437                                                                    ; LOWER BYTE
7438  016436  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC=11
7439  016442  022700  177600                    CMP     #177600,R0      ; CHECK FOR SIGN EXTENSION IN R0
7440  016446  001404                            BEQ     1$
7441  016450  012745  000354                    MOV     #354,-(R5)
 (2)  016454  005245                            INC     -(R5)
 (2)  016456  000000                            HALT                    ; SIGN WAS NOT EXTENDED IN R0
7442  016460  000277                    1$:     SCC
7443  016462  012700  177777                    MOV     #177777,R0
7444  016466  112700  000000                   MOVB    #0,R0           ;CLEAR THE LOWER BYTE OF R0.
7445  016472  004737  017206                    JSR     PC,@#$CC5       ; CHECK FOR CC=5
7446  016476  005700                            TST     R0              ; CHECK R0 FOR SIGN EXTENTION
7447  016500  001404                            BEQ     2$
7448  016502  012745  000355                    MOV     #355,-(R5)
 (2)  016506  005245                            INC     -(R5)
 (2)  016510  000000                            HALT                    ; SIGN WAS NOT EXTENDED IN R0.
7449  016512  012704  000444            2$:     MOV     #TEMP2,R4       ; R4 IS POINTING TO TEMP2
7450  016516  012714  000377                    MOV     #377,(R4)       ; PLACE #377 IN LOCATION TEMP2
7451  016522  012706  000526                    MOV     #START-2,R6
7452  016526  116426  000000                   MOVB    0(R4),(R6)+     ; PUSH # 377 ON STACK
7453  016532  022706  000530                    CMP     #START,R6
7454  016536  001404                            BEQ     3$
7455  016540  012745  000356                    MOV     #356,-(R5)
 (2)  016544  005245                            INC     -(R5)
 (2)  016546  000000                            HALT                    ; R6 DID NOT GET INCREMENTED
7456                                                                    ; BY 2 BY A BYTE INSTRUCTION
7457  016550  124627  000377            3$:     CMPB    -(R6),#377      ; CHECK LOCATION START-2 TO
7458                                                                    ; CONTAIN PROPER DATA
7459  016554  001404                            BEQ     4$
7460  016556  012745  000357                    MOV     #357,-(R5)
 (2)  016562  005245                            INC     -(R5)
 (2)  016564  000000                            HALT                    ; BYTE INSTRUCTION IS FAILING WITH R6
7461  016566  022706  000526            4$:     CMP     #START-2,R6     ; CHECK THAT R6 WAS DECREMENTED
7462                                                                    ; BY 2 BY A BYTE INSTRUCTION
7463  016572  001404                            BEQ     5$
7464  016574  012745  000360                    MOV     #360,-(R5)
 (2)  016600  005245                            INC     -(R5)
 (2)  016602  000000                            HALT                    ; R6 WAS NOT DECREMENTED
7465  016604  016467  000000  161626    5$:     MOV     0(R4),TEMP      ; SET THE LOWER BYTE OF LOCATION TEMP
7466  016612  005726                            TST     (R6)+           ; RESTORE STACK POINTER
7467  016614  000277                            SCC
7468  016616  114667  161617                   MOVB    -(SP),TEMP+1    ; SET THE HIGHER BYTE OF LOCATION TEMP
7469  016622  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC=11
7470  016626  022767  177777  161604          CMP     #177777,TEMP    ; CHECK TEMP FOR THE CORRECT VALUE
7471  016634  001404                            BEQ     6$
```

N07


```
  7472  016636  012745  000361                    MOV     #361,-(R5)
   (2)  016642  005245                            INC     -(R5)
   (2)  016644  000000                            HALT                    ; TEMP FOULED UP
  7473  016646  005067  161566          6$:       CLR     TEMP
  7474  016652  000241                            CLC
  7475  016654  105167  161561                    COMB    TEMP+1          ; WRITE 1'S IN THE HIGHER BYTE OF TEMP
  7476  016660  004737  017272                    JSR     PC,@#$CC11      ; CHECK FOR CC=11
  7477  016664  022767  177400  161546          CMP     #177400,TEMP
  7478  016672  001404                            BEQ     NEXT
  7479  016674                            EBTWRD:
   (2)  016674  012745  000362                    MOV     #362,-(R5)
   (2)  016700  005245                            INC     -(R5)
   (2)  016702  000000                            HALT                    ; WRONG VALUE IN TEMP OR WRONG SEQUENCE
```

```
7484
7485
7486
7487                                    ;               END OF PASS
7488                                    ;               ***********
7489
7490
7491
7492  016704                            NEXT:
 (2)  016704  021527  000123                       CMP     (R5),#123
7493  016710  001404                               BEQ     2$              ; IF IN WRONG SEQUENCE GO TO HLT BELOW
7494  016712  012745  000363                       MOV     #363,-(R5)
 (2)  016716  005245                               INC     -(R5)
 (2)  016720  000000                               HALT                    ; PROGRAM IS IN WRONG SEQUENCE
7495  016722  005267  161460            2$:       INC     $PASS
7496  016726  126727  161454  000001              CMPB    $PASS,#1        ; ALLOW THE TYPE OUT OF END OF
7497                                                                        ; PASS EVERY 377 PASSES
7498  016734  001002                               BNE     GET42
7499  016736  000004  000454                       .TYPE   ,ENDPAS         ; TYPE END OF PASS MESSAGE
7500  016742  013700  000042            GET42:    MOV     @#42,R0
7501  016746  001404                               BEQ     DOAGN
7502  016750  004710                    $ENDAD:   JSR     PC,(R0)
7503  016752  000240                               NOP
7504  016754  000240                               NOP
7505  016756  000240                               NOP
7506  016760  005067  161420            DOAGN:    CLR     $TESTN          ; PREPARE TO START FROM TEST 0
7507  016764  000167  161540            RETURN:   JMP     START           ; START TEST OVER AT BEGINNING
7508                                    ;         ------------------------
7509
7510                                    ;;*************************************************************************
7511
7512                                    .SBTTL  POWER FAIL ROUTINE
7513
7514
7515  016770  012737  017000  000024    PWRDN:    MOV     #PWRUP,@#24     ; GO TO POWER UP ROUTINE AFTER THE POWER COMES BACK
7516  016776  000000                               HALT
7517
7518  017000  012706  000530            PWRUP:    MOV     #START,SP
7519  017004  012737  016770  000024              MOV     #PWRDN,@#24
7520  017012  000004  000470                       .TYPE   ,POWER
7521  017016  000760                               BR      DOAGN
```

```
                                        .SBTTL  TYPE ROUTINE

017020  132737  000040  000421  TYPE:   BITB    #40,@#$ENVM     ; HAS THE CONSOLE OUTPUTS BEEN SUPPRESSED?
017026  001012                          BNE     4$              ; IF SO THEN GO TO 4$
017030  017603  000000                  MOV     @(SP),R3        ; GET ADDRESS OF MESSAGE

017034  105713                  1$:     TSTB    (R3)            ; END OF MESSAGE ?
017036  001406                          BEQ     4$              ; YES, GO WRAP IT UP

017040  105777  161402          3$:     TSTB    @TPS            ; READY FOR NEXT CHARACTER ?
017044  100375                          BPL     3$              ; NO, WAIT
017046  112377  161376                  MOVB    (R3)+,@TPB      ; LOAD AND TYPE THE CHARACTER
017052  000770                          BR      1$              ; YES, GET THE NEXT CHARACTER

017054  062716  000002          4$:     ADD     #2,(SP)         ; ADJUST THE RETURN PC
017060  000006                          RTT                     ; RETURN
```

```
    017062  003402          $CCO:   BLE     1$
    017064  100401                  BMI     1$
    017066  103004                  BCC     2$
    017070                  1$:
(2) 017070  012745  000364          MOV     #364,-(R5)
(2) 017074  005245                  INC     -(R5)
(2) 017076  000000                  HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 0
    017100  000207          2$:     RTS     PC

    017102  003402          $CC1:   BLE     1$
    017104  100401                  BMI     1$
    017106  103404                  BCS     2$
    017110                  1$:
(2) 017110  012745  000365          MOV     #365,-(R5)
(2) 017114  005245                  INC     -(R5)
(2) 017116  000000                  HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 1
    017120  000207          2$:     RTS     PC

    017122  100402          $CC2:   BMI     1$
    017124  101401                  BLOS    1$
    017126  102404                  BVS     2$
    017130                  1$:
(2) 017130  012745  000366          MOV     #366,-(R5)
(2) 017134  005245                  INC     -(R5)
(2) 017136  000000                  HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 2
    017140  000207          2$:     RTS     PC

    017142  100403          $CC3:   BMI     1$
    017144  001402                  BEQ     1$
    017146  102001                  BVC     1$
    017150  103404                  BCS     2$
    017152                  1$:
(2) 017152  012745  000367          MOV     #367,-(R5)
(2) 017156  005245                  INC     -(R5)
(2) 017160  000000                  HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 3
    017162  000207          2$:     RTS     PC

    017164  100403          $CC4:   BMI     1$
    017166  001002                  BNE     1$
    017170  102401                  BVS     1$
    017172  103004                  BCC     2$
    017174                  1$:
(2) 017174  012745  000370          MOV     #370,-(R5)
(2) 017200  005245                  INC     -(R5)
(2) 017202  000000                  HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 4
    017204  000207          2$:     RTS     PC

    017206  100403          $CC5:   BMI     1$
    017210  001002                  BNE     1$
    017212  102401                  BVS     1$
    017214  103404                  BCS     2$
    017216                  1$:
(2) 017216  012745  000371          MOV     #371,-(R5)
(2) 017222  005245                  INC     -(R5)
(2) 017224  000000                  HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 5
    017226  000207          2$:     RTS     PC
```

```
 7585
 7586
 7587  017230  100403                $CC7:   BMI     1$
 7588  017232  001002                        BNE     1$
 7589  017234  102001                        BVC     1$
 7590  017236  103404                        BCS     2$
 7591  017240                        1$:
  (2)  017240  012745  000372                MOV     #372,-(R5)
  (2)  017244  005245                        INC     -(R5)
  (2)  017246  000000                        HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 7
 7592  017250  000207                2$:     RTS     PC
 7593
 7594  017252  100002                $CC10:  BPL     1$
 7595  017254  101401                        BLOS    1$
 7596  017256  102004                        BVC     2$
 7597  017260                        1$:
  (2)  017260  012745  000373                MOV     #373,-(R5)
  (2)  017264  005245                        INC     -(R5)
  (2)  017266  000000                        HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 10
 7598  017270  000207                2$:     RTS     PC
 7599
 7600  017272  100003                $CC11:  BPL     1$
 7601  017274  001402                        BEQ     1$
 7602  017276  102401                        BVS     1$
 7603  017300  103404                        BCS     2$
 7604  017302                        1$:
  (2)  017302  012745  000374                MOV     #374,-(R5)
  (2)  017306  005245                        INC     -(R5)
  (2)  017310  000000                        HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 11
 7605  017312  000207                2$:     RTS     PC
 7606
 7607  017314  100002                $CC12:  BPL     1$
 7608  017316  101401                        BLOS    1$
 7609  017320  102404                        BVS     2$
 7610  017322                        1$:
  (2)  017322  012745  000375                MOV     #375,-(R5)
  (2)  017326  005245                        INC     -(R5)
  (2)  017330  000000                        HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 12
 7611  017332  000207                2$:     RTS     PC
 7612
 7613  017334  100002                $CC13:  BPL     1$
 7614  017336  003401                        BLE     1$
 7615  017340  103404                        BCS     2$
 7616  017342                        1$:
  (2)  017342  012745  000376                MOV     #376,-(R5)
  (2)  017346  005245                        INC     -(R5)
  (2)  017350  000000                        HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 13
 7617  017352  000207                2$:     RTS     PC
 7618
 7619  017354  100003                $CC17:  BPL     1$
 7620  017356  001002                        BNE     1$
 7621  017360  102001                        BVC     1$
 7622  017362  103404                        BCS     2$
 7623  017364                        1$:
  (2)  017364  012745  000377                MOV     #377,-(R5)
  (2)  017370  005245                        INC     -(R5)
```

```
   (2)  017372  000000                          HALT                    ;WRONG CC, IT SHOULD HAVE BEEN = 17
 7624   017374  000207                 2$:      RTS      PC
 7625
 7626          000001                 .END
```

```
ABASE = 000000           5031
ACDW1 = 000000           5031
ACDW2 = 000000           5031
ACPUOP= 000000           5031
ADCB0   004140           5664    5668#
ADCB1   012414           6758    6765#
ADC0    005734           5973    5977#
ADC1    014570           7133    7140#
ADDW0 = 000000           5031
ADDW1 = 000000           5031
ADDW10= 000000           5031
ADDW11= 000000           5031
ADDW12= 000000           5031
ADDW13= 000000           5031
ADDW14= 000000           5031
ADDW15= 000000           5031
ADDW2 = 000000           5031
ADDW3 = 000000           5031
ADDW4 = 000000           5031
ADDW5 = 000000           5031
ADDW6 = 000000           5031
ADDW7 = 000000           5031
ADDW8 = 000000           5031
ADDW9 = 000000           5031
ADD0    006540           6106    6113#
ADD1    015430           7273    7277#
ADEVCT= 000000           5031
ADEVM = 000000           5031
ADR     000430           5035#   5036    6289*   6292    6306    6380*   6383    6404    6454*   6457    6887    6918
ADR1    000432           5037#   5038    6290*   6293    6307    6381*   6384    6405    6455*   6458    6888    6919
ADR2    000434           5039#   5040    6291*   6382*   6385    6406    6456*   6459
AENV  = 000000           5031
AENVM = 000000           5031
AFATAL= 000000           5031
AMADR1= 000000           5031
AMADR2= 000000           5031
AMADR3= 000000           5031
AMADR4= 000000           5031
AMAMS1= 000000           5031
AMAMS2= 000000           5031
AMAMS3= 000000           5031
AMAMS4= 000000           5031
AMSGAD= 000000           5031
AMSGLG= 000000           5031
AMSGTY= 000000           5031
AMTYP1= 000000           5031
AMTYP2= 000000           5031
AMTYP3= 000000           5031
AMTYP4= 000000           5031
APASS = 000000           5031
APRIOR= 000000           5031
ASLB0   003742           5622    5626#
ASLB1   012202           6711    6718#
ASL0    005536           5931    5935#
ASL1    014356           7086    7093#
ASRB0   004032           5645#
```

```
ASRB1   012276          6737#
ASRO    005626          5954#
ASR1    014452          7112#
ASWREG= 000000          5031
ATESTN= 000000          5031
AUNIT = 000000          5031
AUSWR = 000000          5031
AVECT1= 000000          5031
AVECT2= 000000          5031
BICBO   003074          5483    5487#
BICB1   011136          6527    6534#
BICO    004644          5784    5788#
BIC1    013272          6905    6913#
BRANCH  001170          5223    5229#
BTWRD   016416          7426    7431#
CBIT    000740          5149    5154#
CCO     000612          5096    5099    5100    5101    5102    5104    5105    5106    5107    5108    5109    5110    5112#
CC1     000656          5119    5122    5123    5124    5125    5126    5127    5128    5130#
CC2     000726          5135    5139    5140    5141    5142    5143    5144    5145    5146    5148#
CC3     000774          5155    5160    5161    5162    5163    5164    5165    5167#
CC4     001044          5172    5178    5179    5180    5181    5182    5183    5184    5186#
CC5     001156          5210    5214    5216    5218    5220    5222#
CC6     001112          5193    5196    5197    5198    5199    5201    5202    5203    5204    5206#
CLNZ  = 000254          5023#   6048    7214
CMPBO   003000          5469#
CMPB1   011034          6504    6511#
CMPO    004552          5771#
CMP1    013026          6865#
COMBO   003352          5543#
COMB1   011560          6613    6626#
COMO    005146          5851#
COM1    013726          7001#
DOAGN   016760          7501    7506#   7521
DUMMY   000436          5041#   5042    5377    5378
EADDO   006700          6114    6140#
EADD1   015724          7278    7324#
EASRBO  004130          5646    5665#
EASRB1  012404          6738    6759#
EASRO   005724          5955    5974#
EASR1   014560          7113    7134#
EBICO   005000          5789    5815#
EBIC1   013544          6914    6961#
EBTWRD  016674          7432    7479#
ECMPBO  003064          5470    5484#
ECMPB1  011126          6512    6528#
ECMPO   004634          5772    5785#
ECMP1   013262          6866    6906#
EINCB1  011550          6578    6614#
EMODEO  007332          6216    6246#
EMODE2  007470          6254    6279    6282#
EMODE3  007662          6287    6312    6315#
EMODE4  010140          6322    6369    6372#
EMODE5  010370          6377    6416    6419#
EMODE6  010540          6426    6446#
EMODE7  010702          6451    6469#
ENDCCO  000622          5111    5113#
```

```
ENDCC1  000666          5129    5131#
ENDCC2  000736          5147    5149#
ENDCC5  001166          5221    5223#
ENDJP1  001356          5245    5257#
ENDJP3  001562          5263    5289#
ENDJP5  001772          5293    5321#
ENDJP7  002172          5326    5352#
ENDJSR  002606          5356    5415#
ENDPAS  000454          5052#   7499
ENEGB0  003546          5570    5583#
ENEGB1  011776          6655    6669#
ENEGO   005342          5878    5891#
ENEG1   014146          7027    7043#
EPSW    007146          6174    6203#
EPSWNO  016406          7394    7427#
EREGS   002714          5420    5440#
EROLB0  003640          5587    5602#
EROLB1  012074          6676    6692#
EROLO   005434          5895    5910#
EROL1   014250          7050    7067#
ERORB0  003732          5608    5623#
ERORB1  012172          6696    6712#
ERORO   005526          5917    5932#
EROR1   014346          7071    7087#
ERRNM = 000400          5024#   5112#   5130#   5148#   5167#   5186#   5206#   5222#   5231#   5234#   5236#   5238#   5240#
                        5250#   5251#   5254#   5257#   5268#   5269#   5272#   5275#   5282#   5285#   5289#   5298#   5300#
                        5303#   5306#   5312#   5313#   5316#   5321#   5330#   5333#   5336#   5339#   5344#   5348#   5352#
                        5361#   5364#   5365#   5368#   5371#   5375#   5382#   5383#   5386#   5389#   5392#   5393#   5397#
                        5401#   5407#   5412#   5415#   5440#   5453#   5477#   5481#   5484#   5489#   5498#   5503#   5515#
                        5530#   5545#   5553#   5559#   5577#   5583#   5596#   5602#   5616#   5623#   5628#   5637#   5654#
                        5665#   5670#   5683#   5690#   5702#   5710#   5718#   5727#   5748#   5762#   5779#   5782#   5785#
                        5799#   5804#   5815#   5823#   5838#   5853#   5861#   5867#   5885#   5891#   5904#   5910#   5925#
                        5932#   5937#   5946#   5963#   5974#   5979#   5992#   5999#   6011#   6019#   6027#   6036#   6053#
                        6059#   6075#   6083#   6107#   6121#   6127#   6135#   6140#   6145#   6153#   6158#   6165#   6188#
                        6203#   6223#   6229#   6240#   6246#   6270#   6282#   6303#   6315#   6331#   6342#   6349#   6357#
                        6372#   6395#   6402#   6419#   6440#   6446#   6464#   6469#   6502#   6505#   6522#   6525#   6528#
                        6536#   6548#   6553#   6568#   6571#   6601#   6614#   6628#   6639#   6646#   6663#   6669#   6686#
                        6692#   6705#   6712#   6720#   6730#   6748#   6759#   6767#   6781#   6788#   6797#   6806#   6814#
                        6823#   6845#   6875#   6878#   6895#   6899#   6906#   6928#   6933#   6940#   6952#   6961#   6966#
                        6985#   7003#   7012#   7018#   7037#   7043#   7060#   7067#   7080#   7087#   7095#   7105#   7123#
                        7134#   7142#   7156#   7163#   7172#   7181#   7189#   7198#   7219#   7225#   7239#   7247#   7274#
                        7288#   7293#   7300#   7318#   7324#   7341#   7345#   7352#   7361#   7372#   7378#   7381#   7386#
                        7413#   7427#   7441#   7448#   7455#   7460#   7464#   7472#   7479#   7494#   7550#   7556#   7562#
                        7569#   7576#   7583#   7591#   7597#   7604#   7610#   7616#   7623#
ESOB    016264          7365    7386#
ESUB1   016142          7331    7361#
ESWABO  006422          6066    6083#
ESWAB1  015300          7229    7247#
ESXTO   006322          6044    6059#
ESXT1   015172          7209    7225#
ETSTB1  011024          6484    6505#
EXORO   006530          6087    6107#
EXOR1   015420          7254    7274#
GET42   016742          7498    7500#
INCBO   003214          5513#
INCB1   011364          6577#
```

```
INC0     005010            5814    5821#
INC1     013554            6960    6964#
JMP1     001262            5241    5244#
JMP2     001366            5255    5262#
JMP3     001472            5273    5277#
JMP4     001572            5286    5292#
JMP5     001670            5304    5308#
JMP6     002002            5319    5325#
JMP7     002104            5337    5341#
JSRM     002344            5363    5376#
JSRTST   002202            5349    5355#
MARK2    000452            5051#   5379
MODE0    007156            6202    6215#
MODE1    007242            6228    6231#
MODE2    007342            6245    6253#
MODE3    007500            6281    6286#
MODE4    007672            6314    6321#
MODE5    010150            6371    6376#
MODE6    010400            6418    6425#
MODE7    010550            6445    6450#
NBIT     000624            5113    5118#
NEGB0    003466            5569#
NEGB1    011712            6654#
NEG0     005262            5877#
NEG1     014052            7026#
NEXT     016704            7478    7492#
NOBIT    000546            5088    5095#
NOP1   = 000260            5025#   5103
NOTCC    001122            5205    5209#
PC     =%000007            5022#   5335    5360*   5374*   5457*   5459*   5461*   5463*   5474*   5495*   5500*   5505*   5507*
                           5520*   5523*   5525*   5527*   5533*   5535*   5538*   5540*   5550*   5556*   5563*   5574*   5580*
                           5593*   5598*   5618*   5620*   5634*   5639*   5641*   5656*   5658*   5662*   5675*   5680*   5687*
                           5694*   5707*   5715*   5721*   5724*   5731*   5752*   5754*   5757*   5759*   5765*   5776*   5796*
                           5801*   5806*   5809*   5812*   5828*   5831*   5833*   5835*   5841*   5843*   5846*   5848*   5858*
                           5864*   5871*   5882*   5888*   5901*   5906*   5927*   5929*   5943*   5948*   5950*   5965*   5967*
                           5971*   5984*   5989*   5996*   6003*   6016*   6024*   6030*   6033*   6040*   6050*   6056*   6072*
                           6080*   6093*   6099*   6104*   6118*   6124*   6130*   6150*   6162*   6168*   6182*   6189*   6195*
                           6200*   6490*   6492*   6494*   6496*   6519*   6545*   6550*   6556*   6558*   6565*   6574*   6584*
                           6588*   6594*   6598*   6604*   6606*   6609*   6611*   6636*   6642*   6651*   6660*   6666*   6683*
                           6688*   6707*   6709*   6727*   6732*   6734*   6750*   6752*   6756*   6773*   6778*   6785*   6792*
                           6803*   6811*   6817*   6820*   6827*   6851*   6853*   6857*   6859*   6872*   6880*   6882*   6884*
                           6925*   6930*   6937*   6944*   6946*   6958*   6972*   6978*   6980*   6982*   6988*   6990*   6993*
                           6995*   7009*   7015*   7023*   7034*   7040*   7057*   7062*   7082*   7084*   7102*   7107*   7109*
                           7125*   7127*   7131*   7148*   7153*   7160*   7167*   7178*   7186*   7192*   7195*   7202*   7216*
                           7222*   7236*   7244*   7260*   7266*   7271*   7285*   7297*   7304*   7310*   7314*   7321*   7338*
                           7349*   7355*   7358*   7375*   7404*   7409*   7419*   7424*   7438*   7445*   7469*   7476*   7502*
                           7551*   7557*   7563*   7570*   7577*   7584*   7592*   7598*   7605*   7611*   7617*   7624*
POWER    000470            5054#   7520
PSW      007054            6173#
PSWNO    016274            7385    7393#
PWRDN    016770            5074    7515#   7519
PWRUP    017000            7515    7518#
REGS     002616            5396    5419#
RETURN   016764            7507#
ROLB0    003556            5582    5586#
ROLB1    012006            6668    6675#
```

```
ROLD     005352          5890    5894#
ROL1     014156          7042    7049#
RORBO    003650          5601    5607#
RORB1    012104          6691    6695#
RORO     005444          5909    5916#
ROR1     014260          7066    7070#
R0      =%000000         5014#   5075*   5076*   5077    5247*   5249    5252    5255*   5256    5265*   5267    5270    5273*
                         5274    5279*   5281    5283    5287*   5288    5295*   5297    5301    5304*   5305    5309*   5311
                         5314    5318*   5320    5346*   5347    5350*   5351    5391*   5409    5413*   5414*   5423*   5429
                         5456*   5458    5478*   5479    5482    5492*   5494    5496    5501    5572*   5573*   5575    5578*
                         5579*   5581    5672*   5674*   5677*   5679*   5681    5684*   5686*   5688    5691*   5693*   5751*
                         5753    5764*   5774*   5775    5777    5792*   5793*   5795    5797    5807*   5808    5811*   5813
                         5880*   5881*   5883    5886*   5887*   5889    5981*   5983*   5986*   5988*   5990    5993*   5995*
                         5997    6000*   6002*   6095*   6098    6122*   6123*   6125    6177*   6181*   6190*   6194*   6218*
                         6219    6224*   6225    6231*   6235*   6236    6241*   6242    6256*   6260*   6261    6264*   6265
                         6274*   6275*   6276    6292*   6296    6298*   6306*   6308*   6309    6325*   6336*   6338    6339*
                         6340    6344*   6351*   6362    6364    6365    6367*   6383*   6390    6391    6392    6393    6396
                         6397*   6404*   6411    6412    6413    6414*   6429*   6432*   6433*   6436    6437*   6442*   6443*
                         6457*   6460*   6461*   6465*   6466*   6486*   6489*   6491    6497    6540*   6541    6544    6559*
                         6560*   6561*   6563*   6566    6569    6573*   6586*   6588    6606    6609    6643*   6644    6657*
                         6658*   6659*   6661    6664*   6665*   6667    6769*   6770*   6772*   6775*   6777*   6779    6782*
                         6784*   6786    6789*   6791*   6848*   6850*   6852    6854*   6855*   6856    6869*   6870*   6871
                         6892*   6893    6900*   6901    6902    6903*   6921*   6922*   6924    6926    6931    6934*   6935
                         6938    6941*   6943*   6953*   6954*   6955*   6957    6974*   6975*   6978    6990    7019*   7020*
                         7022*   7144*   7145*   7147*   7150*   7152*   7154    7157*   7159*   7161    7164*   7166*   7262*
                         7265*   7268*   7270*   7280*   7283    7284    7289*   7290*   7291    7295    7296    7302    7303
                         7306*   7308    7311*   7313    7367*   7374*   7376    7396*   7399*   7403*   7434*   7436*   7439
                         7443*   7444*   7446    7500*   7502
R1      =%000001         5015#   5390*   5391    5424*   5430    5460*   5462    5589*   5591*   5592*   5594    5597*   5599*
                         5600    5704*   5706*   5708    5712*   5714*   5716    5720*   5723*   5725    5728*   5730*   5756*
                         5758    5760    5897*   5899*   5900*   5902    5905*   5907*   5908    6013*   6015*   6017    6021*
                         6023*   6025    6029*   6032*   6034    6037*   6039*   6116*   6117*   6119    6132*   6133*   6137*
                         6138    6176*   6183*   6196*   6201    6219*   6220    6225*   6226    6232*   6236*   6237    6242*
                         6243    6257*   6261*   6262*   6263    6265*   6272    6277*   6293*   6296*   6297    6307*   6310*
                         6326*   6333*   6335*   6343*   6344    6352*   6359    6361*   6366    6367    6384*   6388    6389*
                         6396    6397    6405*   6409    6410*   6413    6414    6430*   6437    6458*   6487*   6493*   6495
                         6499*   6500    6503    6514*   6516*   6523    6526    6541*   6542*   6546    6551    6678*   6679*
                         6681*   6682*   6684    6687*   6689*   6690    6799*   6800*   6802*   6804    6808*   6810*   6812
                         6816*   6819*   6821    6824*   6826*   6847*   6856*   6858    7052*   7053*   7055*   7056*   7058
                         7061*   7063    7065    7174*   7175*   7177*   7179    7183*   7185*   7187    7191*   7194*   7196
                         7199*   7201*   7267*   7270    7281*   7283*   7284*   7295*   7296*   7302*   7303*   7307*   7309*
                         7368*   7369*   7370    7379    7397*   7398*   7405*   7410    7414*
R2      =%000002         5016#   5425*   5431    5473*   5475    5479    5482    5610*   5612*   5613*   5614    5617*   5619*
                         5621    5775*   5777    5780    5783    5919*   5921*   5922*   5923    5926*   5928*   5930    6046*
                         6049*   6051    6055*   6057    6100*   6103    6128*   6129*   6147*   6149    6155    6160*   6161*
                         6163    6220*   6221    6226*   6227    6233*   6237*   6243*   6258*   6259*   6266    6273*   6278
                         6327*   6328*   6329    6332*   6334*   6335    6337*   6338*   6353*   6354    6355    6358*   6360
                         6361    6363    6364*   6385*   6386    6387*   6388    6389    6390    6391*   6392    6393    6406*
                         6407    6408*   6409    6410    6411    6412*   6431*   6436*   6443    6459*   6461    6466    6497*
                         6498*   6499    6503    6515*   6518*   6520    6523    6526    6698*   6699*   6701*   6702*   6703
                         6706*   6708*   6710    6868*   6871*   6873    6876    6879    6881    6883    6890*   6892    6896
                         6935*   6936    7031*   7032*   7041    7063*   7064*   7073*   7074*   7076*   7077*   7078    7081*
                         7083*   7085    7211*   7212*   7215*   7217    7221*   7223    7333*   7337    7343    7348*   7353*
                         7354*
R3      =%000003         5017#   5328*   5329    5331    5337*   5338    5341*   5342*   5343    5345*   5380*   5383*   5386*
                         5426*   5432    5491*   5494*   5496    5499    5501*   5504*   5506    5547*   5549*   5551    5555*
```

```
                    5557    5560*   5562*   5630*   5632*   5633*   5635    5638*   5640*   5659*   5660*   5661*   5663
                    5791*   5795*   5797    5800    5802*   5805*   5808    5855*   5857*   5859    5863*   5865    5868*
                    5870*   5939*   5941*   5942*   5944    5947*   5949*   5968*   5969*   5970*   5972    6068*   6071*
                    6073    6076*   6079*   6081    6090*   6092*   6094*   6098*   6148*   6149*   6151    6154*   6156*
                    6159*   6161    6167    6538*   6539*   6544*   6546    6549    6551*   6555*   6557    6559    6630*
                    6633*   6635*   6637    6641*   6644    6648*   6650*   6722*   6723*   6725*   6726*   6728    6731*
                    6733*   6741*   6753*   6754*   6755*   6757    6916*   6917*   6920    6924*   6926    6929    6931*
                    6936*   6945    6948*   6949*   6950    6954    6955    7005*   7006*   7008*   7010    7014*   7016
                    7019    7097*   7098*   7100*   7101*   7103    7106*   7108*   7116*   7128*   7129*   7130*   7132
                    7231*   7232*   7235*   7237    7240*   7243*   7245    7334*   7337*   7343*   7348    7354    7530*
                    7532    7537
R4      =%000004    5018#   5381*   5399    5427*   5433    5517*   5519*   5521*   5522*   5524*   5526*   5528    5532*
                    5534*   5536*   5537*   5539*   5648*   5650*   5651*   5652    5655*   5657*   5755*   5756    5760
                    5825*   5827*   5829*   5830*   5832*   5834*   5836    5840*   5842*   5844*   5845*   5847*   5957*
                    5959*   5960*   5961    5964*   5966*   6089*   6092    6095    6101*   6103*   6105    6131*   6133
                    6136*   6138*   6155*   6156    6166*   6167*   6517*   6518    6520    6580*   6581*   6583*   6585*
                    6587*   6589    6591    6595    6597*   6599    6603*   6605*   6607*   6608*   6610*   6612    6631*
                    6632*   6633    6643    6647*   6648    6740*   6742*   6744*   6745*   6746    6749*   6751*   6887*
                    6888*   6889*   6890    6918*   6919*   6920*   6945    6948    6968*   6969*   6971*   6973*   6977*
                    6979*   6981*   6983    6987*   6989*   6991*   6992*   6994*   7029*   7030*   7031    7033*   7035
                    7038*   7039*   7041    7115*   7117*   7119*   7120*   7121    7124*   7126*   7256*   7259    7265
                    7308*   7309    7382*   7383*   7384    7449*   7450*   7452    7465
R5      =%000005    5019#   5086*   5087    5095    5097*   5112*   5118    5120*   5130*   5134    5136*   5148*   5154
                    5156*   5167*   5171    5173*   5186*   5192    5194*   5206*   5209    5211*   5222*   5229    5231*
                    5232*   5234*   5236*   5238*   5240*   5244    5246*   5250*   5251*   5254*   5257*   5262    5264*
                    5268*   5269*   5272*   5275*   5282*   5285*   5289*   5292    5294*   5298*   5300*   5303*   5306*
                    5312*   5313*   5316*   5321*   5325    5327*   5330*   5333*   5336*   5339*   5344*   5348*   5352*
                    5355    5357*   5361*   5364*   5365*   5368*   5371*   5375*   5376    5380    5382*   5384    5389*
                    5392*   5393*   5397*   5401*   5402*   5403    5406*   5407*   5412*   5415*   5419    5421*   5434
                    5440*   5451    5453*   5454*   5469    5471*   5477*   5481*   5484*   5487    5489*   5490*   5498*
                    5503*   5513    5515*   5516*   5530*   5543    5545*   5546*   5553*   5559*   5569    5571*   5577*
                    5583*   5586    5588*   5596*   5602*   5607    5609*   5616*   5623*   5626    5628*   5629*   5637*
                    5645    5647*   5654*   5665*   5668    5670*   5671*   5683*   5690*   5700    5702*   5703*   5710*
                    5718*   5727*   5746    5748*   5749*   5762*   5771    5773*   5779*   5782*   5785*   5788    5790*
                    5799*   5804*   5815*   5821    5823*   5824*   5838*   5851    5853*   5854*   5861*   5867*   5877
                    5879*   5885*   5891*   5894    5896*   5904*   5910*   5916    5918*   5925*   5932*   5935    5937*
                    5938*   5946*   5954    5956*   5963*   5974*   5977    5979*   5980*   5992*   5999*   6009    6011*
                    6012*   6019*   6027*   6036*   6043    6045*   6053*   6059*   6065    6067*   6075*   6083*   6086
                    6088*   6107*   6113    6115*   6121*   6127*   6135*   6140*   6143    6145*   6146*   6153*   6158*
                    6165*   6173    6175*   6188*   6203*   6215    6217*   6223*   6229*   6240*   6246*   6253    6255*
                    6270*   6282*   6286    6288*   6303*   6315*   6321    6323*   6331*   6342*   6349*   6357*   6372*
                    6376    6378*   6395*   6402*   6419*   6425    6427*   6440*   6446*   6450    6452*   6464*   6469*
                    6483    6485*   6502*   6505*   6511    6513*   6522*   6525*   6528*   6534    6536*   6537*   6548*
                    6553*   6568*   6571*   6577    6579*   6601*   6614*   6626    6628*   6629*   6639*   6646*   6654
                    6656*   6663*   6669*   6675    6677*   6686*   6692*   6695    6697*   6705*   6712*   6718    6720*
                    6721*   6730*   6737    6739*   6748*   6759*   6765    6767*   6768*   6781*   6788*   6795    6797*
                    6798*   6806*   6814*   6823*   6843    6845*   6846*   6865    6867*   6875*   6878*   6895*   6899*
                    6906*   6913    6915*   6928*   6933*   6940*   6952*   6961*   6964    6966*   6967*   6985*   7001
                    7003*   7004*   7012*   7018*   7026    7028*   7037*   7043*   7049    7051*   7060*   7067*   7070
                    7072*   7080*   7087*   7093    7095*   7096*   7105*   7112    7114*   7123*   7134*   7140    7142*
                    7143*   7156*   7163*   7170    7172*   7173*   7181*   7189*   7198*   7208    7210*   7219*   7225*
                    7228    7230*   7239*   7247*   7253    7255*   7274*   7277    7279*   7288*   7293*   7300*   7318*
                    7324*   7330    7332*   7341*   7345*   7352*   7361*   7364    7366*   7372*   7378*   7381*   7386*
                    7393    7395*   7413*   7427*   7431    7433*   7441*   7448*   7455*   7460*   7464*   7472*   7479*
                    7492    7494*   7550*   7556*   7562*   7569*   7576*   7583*   7591*   7597*   7604*   7610*   7616*
                    7623*
```

```
R6      =%000006        5020#  5422   5428*  5429*  5430*  5431*  5432*  5433*  5434*  5435   5437*  5439*  7451*
                        7452*  7453   7457   7461   7466
SBCB0     004272        5700#
SBCB1     012552        6795#
SBC0      006066        6009#
SBC1      014726        7170#
SENVC   = 000273        5027#  6054   7220
SEVC    = 000263        5026#  5763   5810   6096   6562   6942   6956   7263
SOB       016152        7360   7364#
SP      =%000006        5021#  5085*  5358*  5362   5366   5369   5373*  5376*  5377*  5378*  5379*  5387   5395
                        5402   5404*  6554*  6555   6590   6591*  6593*  6595   6947*  6949   6957   6959   7312*
                        7313*  7315   7468   7518*  7530   7540*
START     000530        5079   5085#  5358   5362   5366   5387   5395   6959   7451   7453   7461   7507   7518
SUB0      006710        6139   6143#
SUB1      015734        7323   7330#
SWAB0     006332        6058   6065#
SWAB1     015202        7224   7228#
SXT0      006244        6043#
SXT1      015110        7208#
TEMP      000440        5043#  5044   5277*  5278*  5279   5286*  5287   5341   5346   5349*  5350   5422*  5437
                        5439   5792   6231   6256   6263*  6268   6276*  6280   6289   6295*  6297*  6301   6305*
                        6309*  6313   6324*  6325   6340   6345   6350*  6351   6368   6379*  6380   6398   6403*
                        6415   6429   6434   6454   6486   6515   6538   6580   6630   6657   6847   6869   6904
                        6916   6974   7116   7144   7174   7281   7286   7298   7333   7335*  7347*  7350   7356*
                        7357*  7359   7396   7465*  7468*  7470   7473*  7475*  7477
TEMP1     000442        5045#  5046   5308*  5309   5310*  5314   5317*  5318   5319*  6232   6257   6290   6326
                        6352   6381   6430   6441*  6444   6453*  6455   6462   6467   6487   6540   6560   6566
                        6631   6678   6698   6722   6740   6848   6868   6876   6886*  6896   6921   6938   6968
                        7005   7029   7211   7257*  7259*  7261*  7262   7272   7305*  7306   7334   7336*  7339
                        7342*  7346*  7397   7420*  7425
TEMP2     000444        5047#  5048   6233   6234*  6238   6244   6258   6291   6294*  6299   6304*  6311   6327
                        6329   6347   6353   6355   6370   6382   6400   6417   6428*  6431   6438   6456   6514
                        6741   6769   6799   6885*  6886   6901*  6953   7052   7073   7097   7115   7231   7280
                        7282*  7291   7294*  7301*  7319*  7320*  7322   7449
TPB       000450        5050#  7537*
TPS       000446        5049#  7535
TSTB0     002724        5438   5451#
TSTB1     010712        6468   6483#
TST0      004450        5746#
TST1      012734        6843#
TYPE      017020        5059   7528#
VBIT      000670        5131   5134#
XOR0      006432        6082   6086#
XOR1      015310        7246   7253#
YESCC     001054        5185   5192#
ZBIT      001004        5166   5171#
$APTHD    000430        5032#  5034
$CC0      017062        5680   5707   5715   5989   6016   6024   6118   6182   6778   6803   6811   7153   7178
                        7186   7285   7404   7547#
$CC1      017102        5495   5500   5527   5540   5550   5796   5801   5835   5848   5858   6545   6550   6598
                        6611   6636   6880   6925   6930   6982   6995   7009   7553#
$CC10     017252        5461   5463   5757   5759   5776   6072   6150   6494   6496   6519   6857   6859   6872
                        6884   7236   7338   7594#
$CC11     017272        5474   5505   5507   5523   5535   5556   5574   5662   5724   5806   5809   5831   5843
                        5864   5882   5971   6033   6056   6099   6104   6124   6200   6556   6558   6586   6609
                        6642   6660   6756   6820   6937   6946   6975   7015   7034   7131   7195   7222   7266
```

```
                        7271    7297    7424    7438    7469    7476    7600#
$CC12   017314          5593    5620    5634    5687    5901    5929    5943    5996    6683    6709    6727    6785    7057
                        7084    7102    7160    7607#
$CC13   017334          5520    5580    5828    5888    6162    6584    6666    6882    6972    7040    7349    7613#
$CC17   017354          6195    7375    7419    7619#
$CC2    017122          5731    6040    6827    7202    7559#
$CC3    017142          5538    5846    6609    6993    7321    7565#
$CC4    017164          5457    5459    5641    5658    5675    5721    5752    5754    5950    5967    5984    6030    6080
                        6168    6189    6490    6492    6574    6594    6734    6752    6773    6817    6851    6853    7109
                        7127    7148    7192    7244    7355    7358    7409    7572#
$CC5    017206          5525    5533    5563    5694    5765    5812    5833    5841    5871    6003    6050    6093    6565
                        6604    6651    6792    6944    6958    6980    6988    7023    7167    7216    7260    7310    7314
                        7445    7579#
$CC7    017230          5598    5618    5639    5656    5906    5927    5948    5965    6130    6688    6707    6732    6750
                        7062    7082    7107    7125    7304    7587#
$CPUOP  000426          5031#
$DEVCT  000410          5031#
$ENDAD  016750          4918    7502#
$ENV    000420          5031#
$ENVM   000421          5031#   7528
$ETABL  000420          5031#   5075
$ETEND  000430          5031#   5032
$FATAL  000402          5031#
$HD   = 000003          4898
$HIBTS  000430          5032#
$MAIL   000400          5031#   5032    5077
$MBADR  000432          5032#
$MSGAD  000414          5031#
$MSGLG  000416          5031#
$MSGTY  000400          5031#
$PASS   000406          5031#   7495*   7496
$PASTM  000436          5032#
$SVPC = 001000          4918#
$SWR  = 160000          4898#
$SWREG  000422          5031#
$TESTN  000404          5031#   5086    5384    5435    7506*
$TN   = 000124          4898#   5028#   5093    5095#   5116    5118#   5132    5134#   5152    5154#   5169    5171#   5190
                        5192#   5207    5209#   5227    5229#   5242    5244#   5260    5262#   5290    5292#   5323    5325#
                        5353    5355#   5417    5419#   5449    5451#   5467    5469#   5485    5487#   5511    5513#   5541
                        5543#   5567    5569#   5584    5586#   5605    5607#   5624    5626#   5643    5645#   5666    5668#
                        5698    5700#   5744    5746#   5769    5771#   5786    5788#   5819    5821#   5849    5851#   5875
                        5877#   5892    5894#   5914    5916#   5933    5935#   5952    5954#   5975    5977#   6007    6009#
                        6041    6043#   6063    6065#   6084    6086#   6111    6113#   6141    6143#   6171    6173#   6213
                        6215#   6251    6253#   6284    6286#   6319    6321#   6374    6376#   6423    6425#   6448    6450#
                        6481    6483#   6509    6511#   6532    6534#   6575    6577#   6624    6626#   6652    6654#   6673
                        6675#   6693    6695#   6716    6718#   6735    6737#   6763    6765#   6793    6795#   6841    6843#
                        6863    6865#   6911    6913#   6962    6964#   6999    7001#   7024    7026#   7047    7049#   7068
                        7070#   7091    7093#   7110    7112#   7138    7140#   7168    7170#   7206    7208#   7226    7228#
                        7251    7253#   7275    7277#   7328    7330#   7362    7364#   7390    7393#   7429    7431#   7492#
$TSTM   000434          5032#
$UNIT   000412          5031#
$UNITM  000440          5032#
$USWR   000424          5031#
.     = 017376          4906#   4916    4918#   5013#   5032#   5034#   5036#   5038#   5040#   5042#   5044#   5046#   5048#
                        5056#   5058#   5073#   5083#   5335    6181#   6183#   6194#   6196#   7403#   7405#   7418#   7420#
.TYPE = 000004          5029#   7499    7520
```

B09


```
.$X    = 000430          5032#
..A    = 016366          6181#  6183#  6194#  6196#  7403#  7405#  7418#  7420#
..B    = 016372          6181#  6183#  6194#  6196#  7403#  7405#  7418#  7420#
..C    = 000067          6181#  6183#  6194#  6196#  7403#  7405#  7418#  7420#
```

```
COMMEN   974#
ENDCOM   986#
ERROR   4976#  5112  5130  5148  5167  5186  5206  5222  5231  5234  5236  5238  5240  5250  5251
        5254   5257  5268  5269  5272  5275  5282  5285  5289  5298  5300  5303  5306  5312  5313
        5316   5321  5330  5333  5336  5339  5344  5348  5352  5361  5364  5365  5368  5371  5375
        5382   5383  5386  5389  5392  5393  5397  5401  5407  5412  5415  5440  5453  5477  5481
        5484   5489  5498  5503  5515  5530  5545  5553  5559  5577  5583  5596  5602  5616  5623
        5628   5637  5654  5665  5670  5683  5690  5702  5710  5718  5727  5748  5762  5779  5782
        5785   5799  5804  5815  5823  5838  5853  5861  5867  5885  5891  5904  5910  5925  5932
        5937   5946  5963  5974  5979  5992  5999  6011  6019  6027  6036  6053  6059  6075  6083
        6107   6121  6127  6135  6140  6145  6153  6158  6165  6188  6203  6223  6229  6240  6246
        6270   6282  6303  6315  6331  6342  6349  6357  6372  6395  6402  6419  6440  6446  6464
        6469   6502  6505  6522  6525  6536  6548  6553  6568  6571  6601  6614  6628  6639  6646
        6646   6663  6669  6686  6692  6705  6712  6720  6730  6748  6759  6767  6781  6788  6797
        6806   6814  6823  6845  6875  6878  6895  6899  6906  6928  6933  6940  6952  6961  6966
        6985   7003  7012  7018  7037  7043  7060  7067  7080  7087  7095  7105  7123  7134  7142
        7156   7163  7172  7181  7189  7198  7219  7225  7239  7247  7274  7288  7293  7300  7318
        7324   7341  7345  7352  7361  7372  7378  7381  7386  7413  7427  7441  7448  7455  7460
        7464   7472  7479  7494  7550  7556  7562  7569  7576  7583  7591  7597  7604  7610  7616
        7623
ESCAPE  1097#
HLT     4961#  5112  5130  5148  5167  5186  5206  5222  5231  5234  5236  5238  5240  5250  5251
        5254   5257  5268  5269  5272  5275  5282  5285  5289  5298  5300  5303  5306  5312  5313
        5316   5321  5330  5333  5336  5339  5344  5348  5352  5361  5364  5365  5368  5371  5375
        5382   5389  5392  5393  5397  5401  5407  5412  5415  5440  5453  5477  5481  5484  5489
        5498   5503  5515  5530  5545  5553  5559  5577  5583  5596  5602  5616  5623  5628  5637
        5654   5665  5670  5683  5690  5702  5710  5718  5727  5748  5762  5779  5782  5785  5799
        5804   5815  5823  5838  5853  5861  5867  5885  5891  5904  5910  5925  5932  5937  5946
        5963   5974  5979  5992  5999  6011  6019  6027  6036  6053  6059  6075  6083  6107  6121
        6127   6135  6140  6145  6153  6158  6165  6188  6203  6223  6229  6240  6246  6270  6282
        6303   6315  6331  6342  6349  6357  6372  6395  6402  6419  6440  6446  6464  6469  6502
        6505   6522  6525  6528  6536  6548  6553  6568  6571  6601  6614  6628  6639  6646  6663
        6669   6686  6692  6705  6712  6720  6730  6748  6759  6767  6781  6788  6797  6806  6814
        6823   6845  6875  6878  6895  6899  6906  6928  6933  6940  6952  6961  6966  6985  7003
        7012   7018  7037  7043  7060  7067  7080  7087  7095  7105  7123  7134  7142  7156  7163
        7172   7181  7189  7198  7219  7225  7239  7247  7274  7288  7293  7300  7318  7324  7341
        7345   7352  7361  7372  7378  7381  7386  7413  7427  7441  7448  7455  7460  7464  7472
        7479   7494  7550  7556  7562  7569  7576  7583  7591  7597  7604  7610  7616  7623
HLT1    4969#  5383  5386
MFPS    4948#  6183  6196  7405  7420
MTPS    4933#  6181  6194  7403  7418
MULT    3466#
NEWTST  1030#
NWTEST  4982#  5093  5116  5132  5152  5169  5190  5207  5227  5242  5260  5290  5323  5353  5417
        5449   5467  5485  5511  5541  5567  5584  5605  5624  5643  5666  5698  5744  5769  5786
        5819   5849  5875  5892  5914  5933  5952  5975  6007  6041  6063  6084  6111  6141  6171
        6213   6251  6284  6319  6374  6423  6448  6481  6509  6532  6575  6624  6652  6673  6693
        6716   6735  6763  6793  6841  6863  6911  6962  6999  7024  7047  7068  7091  7110  7138
        7168   7206  7226  7251  7275  7328  7362  7390  7429
POP     1483#
PUSH    1475#
REPORT  4437#
SEQCHK  5005#  5095  5118  5134  5154  5171  5192  5209  5229  5244  5262  5292  5325  5355  5419
        5451   5469  5487  5513  5543  5569  5586  5607  5626  5645  5668  5700  5746  5771  5788
        5821   5851  5877  5894  5916  5935  5954  5977  6009  6043  6065  6086  6113  6143  6173
        6215   6253  6286  6321  6376  6425  6450  6483  6511  6534  6577  6626  6654  6675  6695
```

```
                6718    6737    6765    6795    6843    6865    6913    6964    7001    7026    7049    7070    7093    7112    7140
                7170    7208    7228    7253    7277    7330    7364    7393    7431    7492
SEQUEN          5001#   5095    5118    5134    5154    5171    5192    5209    5229    5244    5262    5292    5325    5355    5419
                5451    5469    5487    5513    5543    5569    5586    5607    5626    5645    5668    5700    5746    5771    5788
                5821    5851    5877    5894    5916    5935    5954    5977    6009    6043    6065    6086    6113    6143    6173
                6215    6253    6286    6321    6376    6425    6450    6483    6511    6534    6577    6626    6654    6675    6695
                6718    6737    6765    6795    6843    6865    6913    6964    7001    7026    7049    7070    7093    7112    7140
                7170    7208    7228    7253    7277    7330    7364    7393    7431    7492
SETUP            801#
SKIP            1131#
SLASH            926#
STARS            895#   4897#   4908    4918    4998#   5031    5032    5093    5116    5132    5152    5169    5190    5207    5227
                5242    5260    5290    5323    5353    5417    5449    5467    5485    5511    5541    5567    5584    5605    5624
                5643    5666    5698    5744    5769    5786    5819    5849    5875    5892    5914    5933    5952    5975    6007
                6041    6063    6084    6111    6141    6171    6213    6251    6284    6319    6374    6423    6448    6481    6509
                6532    6575    6624    6652    6673    6693    6716    6735    6763    6793    6841    6863    6911    6962    6999
                7024    7047    7068    7091    7110    7138    7168    7206    7226    7251    7275    7328    7362    7390    7429
                7510
TYPBIN          1419#
TYPDEC          1389#
TYPNAM          1177#
TYPNUM          1356#
TYPOCS          1309#
TYPOCT          1272#
TYPTXT          1226#
$C17            4920#   5251    5269    5365    5383
$NWTST          4986#   5093    5116    5132    5152    5169    5190    5207    5227    5242    5260    5290    5323    5353    5417
                5449    5467    5485    5511    5541    5567    5584    5605    5624    5643    5666    5698    5744    5769    5786
                5819    5849    5875    5892    5914    5933    5952    5975    6007    6041    6063    6084    6111    6141    6171
                6213    6251    6284    6319    6374    6423    6448    6481    6509    6532    6575    6624    6652    6673    6693
                6716    6735    6763    6793    6841    6863    6911    6962    6999    7024    7047    7068    7091    7110    7138
                7168    7206    7226    7251    7275    7328    7362    7390    7429
$$ESCA          1110#
$$NEWT          1064#
$$SKIP          1144#
.EQUAT           168#
.HEADE            39#   4897#   4898
.KT11            308#
.SETUP           747#
.SWRHI            81#
.$ACT1          4032#   4897#   4918
.$APTB          4080#   4897#   5031
.$APTH          4350#   4897#   5032
.$APTY          4513#
.$ASTA          4397#
.$CATC           485#
.$CMTA           586#
.$DB2D          3666#
.$DB2O          3790#
.$DIV           3568#
.$EOP           1535#
.$ERRO          1947#
.$ERRT          2140#
.$MULT          3504#
.$POWE          3223#
.$RAND          3286#
```

```
.$RDDE    2919#
.$RDOC    2827#
.$READ    2613#
.$R2AZ    3934#
.$SAVE    2995#
.$SB2D    3751#
.$SB2O    3853#
.$SCOP    1739#
.$SIZE    3348#
.$SUPR    3891#
.$TRAP    3095#
.$TYPB    2528#
.$TYPD    2450#
.$TYPE    2228#
.$TYPO    2353#
.$40CA     515#
```

F09


```
ADC     5983  5986  5988  5995  6002  7147  7150  7152  7159  7166
ADCB    5674  5677  5679  5686  5693  6772  6775  6777  6784  6791
ADD     5429  5430  5431  5432  5433  5434  6117  6123  6129  6133  6138  7284  7290  7296  7303
        7309  7313  7320  7540
ASL     5941  5942  5947  5949  7100  7101  7106  7108
ASLB    5632  5633  5638  5640  6725  6726  6731  6733
ASR     5959  5960  5964  5966  5969  5970  7119  7120  7124  7126  7129  7130
ASRB    5650  5651  5655  5657  5660  5661  6744  6745  6749  6751  6754  6755
BCC     5181  5199  5204  7549  7575
BCS     5099  5104  5126  5142  5214  5251  5269  5365  5383  7555  7568  7582  7590  7603  7615
        7622
BEQ     5088  5101  5106  5123  5140  5160  5185  5218  5230  5253  5271  5284  5302  5315  5332
        5363  5367  5370  5385  5388  5396  5400  5411  5452  5476  5488  5497  5514  5529  5544
        5552  5558  5576  5582  5595  5601  5615  5622  5627  5636  5653  5664  5669  5682  5689
        5701  5709  5717  5726  5747  5761  5778  5798  5814  5822  5837  5852  5860  5866  5884
        5890  5903  5909  5924  5931  5936  5945  5962  5973  5978  5991  5998  6010  6018  6026
        6035  6052  6058  6074  6082  6106  6120  6126  6134  6139  6144  6152  6157  6164  6187
        6202  6222  6228  6239  6245  6269  6281  6302  6314  6330  6341  6348  6356  6371  6394
        6401  6418  6439  6445  6463  6468  6501  6504  6521  6535  6547  6567  6570  6600  6613
        6627  6638  6645  6662  6668  6685  6691  6704  6711  6719  6729  6747  6758  6766  6780
        6787  6796  6805  6813  6822  6844  6874  6877  6894  6898  6905  6927  6939  6951  6960
        6965  6984  7002  7011  7017  7036  7042  7059  7066  7079  7086  7094  7104  7122  7133
        7141  7155  7162  7171  7180  7188  7197  7218  7224  7238  7246  7273  7287  7292  7299
        7317  7323  7340  7344  7351  7360  7377  7380  7385  7412  7426  7440  7447  7454  7459
        7463  7471  7478  7493  7501  7533  7566  7601
BGE     5124  5149  5166
BGT     5125  5147  5183
BHI     5078  5111  5165  5184  5221
BIC     5795  5805  5811  6924  6936  6943  6949
BICB    5494  5504  6544  6555  6563  6573
BIS     5775  5802  6871  6890  6903  6931
BISB    5473  5501  6518  6551
BIT     5797  5800  5808  6926  6929  6945  6957
BITB    5496  5499  5506  6546  6549  6557  7528
BLE     5109  5129  5144  5164  7371  7547  7553  7614
BLO     5128  5146
BLOS    5110  5127  5145  5205  7560  7595  7608
BLT     5108  5143  5163  5182
BMI     5102  5107  5220  5483  5502  5784  5803  6527  6552  6932  7548  7554  7559  7565  7572
        7579  7587
BNE     5096  5119  5131  5135  5155  5172  5178  5193  5197  5202  5210  5245  5251  5263  5269
        5293  5326  5356  5365  5383  5420  5436  5470  5570  5587  5608  5646  5772  5789  5878
        5895  5917  5955  6044  6066  6087  6114  6174  6216  6254  6267  6279  6287  6300  6312
        6322  6346  6369  6377  6399  6416  6426  6435  6451  6484  6512  6578  6655  6676  6696
        6738  6866  6914  7027  7050  7071  7113  7209  7229  7254  7278  7331  7365  7394  7432
        7498  7529  7573  7580  7588  7620
BPL     5122  5141  5161  5179  5196  5201  5223  5251  5269  5365  5383  5480  5781  6524  7536
        7594  7600  7607  7613  7619
BR      5233  5235  5237  5239  5241  5299  5438  7521  7538
BVC     5113  5139  5162  5180  5198  5203  5251  5269  5365  5383  7567  7589  7596  7621
BVS     5100  5105  5216  7561  7574  7581  7602  7609
CCC     5098  5590  5611  5631  5649  5673  5705  5898  5920  5940  5958  5982  6014  6572  6680
        6700  6724  6743  6771  6801  7054  7075  7099  7118  7146  7176
CLC     5213  6592  7474
CLN     5219  6070  7234
CLR     5076  5428  5751  5981  6046  6177  6181  6183  6194  6196  6234  6273  6304  6350  6403
```

```
                6428    6441    6453    6850    6900    7145    7212    7289    7368    7399    7403    7405    7414    7418    7420
                7434    7473    7506
CLRB            5456    5672    6259    6294    6324    6379    6489    6770
CLV             5215
CLZ             5217    6078    6097    7242    7264
CMP             5077    5095    5118    5134    5154    5171    5192    5209    5229    5244    5252    5262    5270    5283    5292
                5301    5314    5325    5331    5355    5362    5366    5369    5384    5387    5395    5399    5409    5419    5435
                5451    5469    5487    5513    5543    5569    5586    5607    5626    5645    5668    5700    5746    5760    5771
                5777    5780    5783    5788    5821    5836    5851    5859    5865    5877    5883    5889    5894    5902    5908
                5916    5923    5930    5935    5944    5954    5961    5972    5977    5990    5997    6009    6017    6025    6034
                6043    6057    6065    6073    6081    6086    6105    6113    6119    6125    6143    6151    6163    6173    6201
                6215    6227    6244    6253    6278    6286    6311    6321    6329    6340    6355    6368    6376    6388    6390
                6392    6396    6409    6411    6413    6415    6425    6434    6438    6444    6450    6467    6483    6500    6503
                6511    6534    6566    6577    6626    6654    6675    6695    6718    6737    6765    6795    6843    6865    6873
                6876    6879    6881    6883    6893    6896    6902    6904    6913    6938    6950    6959    6964    6983    7001
                7010    7016    7026    7035    7041    7049    7058    7065    7070    7078    7085    7093    7103    7112    7121
                7132    7140    7154    7161    7170    7179    7187    7196    7208    7223    7228    7237    7245    7253    7272
                7277    7286    7291    7298    7322    7330    7339    7350    7364    7370    7379    7393    7410    7425    7431
                7439    7453    7461    7470    7477    7492
CMPB            5475    5479    5482    5528    5551    5557    5575    5581    5594    5600    5614    5621    5635    5652    5663
                5681    5688    5708    5716    5725    6221    6238    6266    6299    6345    6393    6398    6462    6520    6523
                6526    6595    6599    6612    6637    6644    6661    6667    6684    6690    6703    6710    6728    6746    6757
                6779    6786    6804    6812    6821    7457    7496
COM             5857    5863    5870    7008    7014    7022
COMB            5549    5555    5562    6635    6641    6650    7475
DEC             5840    5842    5845    5847    6987    6989    6992    6994
DECB            5532    5534    5537    5539    6603    6605    6608    6610
HALT            4916    5090    5112    5130    5148    5167    5186    5206    5222    5231    5234    5236    5238    5240    5250
                5251    5254    5257    5268    5269    5272    5275    5282    5285    5289    5298    5300    5303    5306    5312
                5313    5316    5321    5330    5333    5336    5339    5344    5348    5352    5361    5364    5365    5368    5371
                5375    5382    5383    5386    5389    5392    5393    5397    5401    5407    5412    5415    5440    5453    5477
                5481    5484    5489    5498    5503    5515    5530    5545    5553    5559    5577    5583    5596    5602    5616
                5623    5628    5637    5654    5665    5670    5683    5690    5702    5710    5718    5727    5748    5762    5779
                5782    5785    5799    5804    5815    5823    5838    5853    5861    5867    5885    5891    5904    5910    5925
                5932    5937    5946    5963    5974    5979    5992    5999    6011    6019    6027    6036    6053    6055    6075
                6083    6107    6121    6127    6135    6140    6145    6153    6158    6165    6188    6203    6223    6229    6240
                6246    6270    6282    6303    6315    6331    6342    6349    6357    6372    6395    6402    6419    6440    6446
                6464    6469    6502    6505    6522    6525    6528    6536    6548    6553    6568    6571    6601    6614    6628
                6639    6646    6663    6669    6686    6692    6705    6712    6720    6730    6748    6759    6767    6781    6788
                6797    6806    6814    6823    6845    6875    6878    6895    6899    6906    6928    6933    6940    6952    6961
                6966    6985    7003    7012    7018    7037    7043    7060    7067    7080    7087    7095    7105    7123    7134
                7142    7156    7163    7172    7181    7189    7198    7219    7225    7239    7247    7274    7288    7293    7300
                7318    7324    7341    7345    7352    7361    7372    7378    7381    7386    7413    7427    7441    7448    7455
                7460    7464    7472    7479    7494    7516    7550    7556    7562    7569    7576    7583    7591    7597    7604
                7610    7616    7623
INC             5097    5112    5120    5130    5136    5148    5156    5167    5173    5186    5194    5206    5211    5222    5231
                5232    5234    5236    5238    5240    5246    5250    5251    5254    5257    5264    5268    5269    5272    5275
                5282    5285    5289    5294    5298    5300    5303    5306    5312    5313    5316    5321    5327    5330    5333
                5336    5339    5344    5348    5352    5357    5361    5364    5365    5368    5371    5375    5382    5383    5386
                5389    5392    5393    5397    5401    5407    5412    5415    5421    5440    5453    5454    5471    5477    5481
                5484    5489    5490    5498    5503    5515    5516    5530    5545    5546    5553    5559    5571    5577    5583
                5588    5596    5602    5609    5616    5623    5628    5629    5637    5647    5654    5665    5670    5671    5683
                5690    5702    5703    5710    5718    5727    5748    5749    5762    5773    5779    5782    5785    5790    5799
                5804    5815    5823    5824    5827    5830    5832    5834    5838    5853    5854    5861    5867    5879    5885
                5891    5896    5904    5910    5918    5925    5932    5937    5938    5946    5956    5963    5974    5979    5980
                5992    5999    6011    6012    6019    6027    6036    6045    6053    6059    6067    6075    6083    6088    6107
```

	6115	6121	6127	6135	6140	6145	6146	6153	6158	6165	6175	6188	6203	6217	6223
	6229	6240	6246	6255	6270	6282	6288	6303	6315	6323	6328	6331	6333	6334	6336
	6337	6342	6349	6357	6372	6378	6395	6402	6419	6427	6440	6446	6452	6464	6469
	6485	6502	6505	6513	6522	6525	6528	6536	6537	6548	6553	6568	6571	6579	6601
	6614	6628	6629	6639	6646	6656	6663	6669	6677	6686	6692	6697	6705	6712	6720
	6721	6730	6739	6748	6759	6767	6768	6781	6788	6797	6798	6806	6814	6823	6845
	6846	6867	6875	6878	6895	6899	6906	6915	6928	6933	6940	6952	6961	6966	6967
	6971	6977	6979	6981	6985	7003	7004	7012	7018	7028	7037	7043	7051	7060	7067
	7072	7080	7087	7095	7096	7105	7114	7123	7134	7142	7143	7156	7163	7172	7173
	7181	7189	7198	7210	7219	7225	7230	7239	7247	7255	7274	7279	7288	7293	7300
	7318	7324	7332	7341	7345	7352	7361	7366	7369	7372	7378	7381	7386	7395	7413
	7427	7433	7441	7448	7455	7460	7464	7472	7479	7494	7495	7550	7556	7562	7569
	7576	7583	7591	7597	7604	7610	7616	7623							
INCB	5519	5522	5524	5526	6262	6264	6339	6343	6583	6587	6593	6597			
IOT	5029														
JMP	5079	5249	5256	5267	5274	5281	5288	5297	5305	5311	5320	5329	5335	5338	5343
	5347	5351	7507												
JSR	5360	5381	5391	5457	5459	5461	5463	5474	5495	5500	5505	5507	5520	5523	5525
	5527	5533	5535	5538	5540	5550	5556	5563	5574	5580	5593	5598	5618	5620	5634
	5639	5641	5656	5658	5662	5675	5680	5687	5694	5707	5715	5721	5724	5731	5752
	5754	5757	5759	5765	5776	5796	5801	5806	5809	5812	5828	5831	5833	5835	5841
	5843	5846	5848	5858	5864	5871	5882	5888	5901	5906	5927	5929	5943	5948	5950
	5965	5967	5971	5984	5989	5996	6003	6016	6024	6030	6033	6040	6050	6056	6072
	6080	6093	6099	6104	6118	6124	6130	6150	6162	6168	6182	6189	6195	6200	6490
	6492	6494	6496	6519	6545	6550	6556	6558	6565	6574	6584	6588	6594	6598	6604
	6606	6609	6611	6636	6642	6651	6660	6666	6683	6688	6707	6709	6727	6732	6734
	6750	6752	6756	6773	6778	6785	6792	6803	6811	6817	6820	6827	6851	6853	6857
	6859	6872	6880	6882	6884	6925	6930	6937	6944	6946	6958	6972	6978	6980	6982
	6988	6990	6993	6995	7009	7015	7023	7034	7040	7057	7062	7082	7084	7102	7107
	7109	7125	7127	7131	7148	7153	7160	7167	7178	7186	7192	7195	7202	7216	7222
	7236	7244	7260	7266	7271	7285	7297	7304	7310	7314	7321	7338	7349	7355	7358
	7375	7404	7409	7419	7424	7438	7445	7469	7476	7502					
MARK	5051														
MOV	5074	5075	5085	5086	5112	5130	5148	5167	5186	5206	5222	5231	5234	5236	5238
	5240	5247	5250	5251	5254	5255	5257	5265	5268	5269	5272	5273	5275	5277	5278
	5279	5282	5285	5286	5287	5289	5295	5298	5300	5303	5304	5306	5308	5309	5310
	5312	5313	5316	5317	5318	5319	5321	5328	5330	5333	5336	5337	5339	5341	5342
	5344	5345	5346	5348	5349	5350	5352	5358	5361	5364	5365	5368	5371	5373	5375
	5376	5377	5378	5379	5380	5382	5383	5386	5389	5390	5392	5393	5397	5401	5402
	5404	5407	5412	5413	5415	5422	5423	5424	5425	5426	5427	5437	5439	5440	5453
	5477	5481	5484	5489	5498	5503	5515	5530	5536	5545	5553	5559	5560	5577	5578
	5583	5596	5602	5616	5623	5628	5637	5654	5665	5670	5683	5690	5702	5710	5718
	5727	5748	5755	5756	5762	5764	5774	5779	5782	5785	5791	5792	5793	5799	5804
	5807	5815	5823	5825	5829	5838	5844	5853	5855	5861	5867	5868	5880	5885	5886
	5891	5897	5904	5910	5919	5925	5932	5937	5939	5946	5957	5963	5968	5974	5979
	5992	5993	5999	6000	6011	6013	6019	6027	6036	6037	6053	6059	6068	6075	6076
	6083	6089	6090	6094	6095	6100	6101	6107	6116	6121	6122	6127	6128	6131	6132
	6135	6137	6140	6145	6147	6148	6153	6154	6155	6158	6159	6160	6165	6166	6176
	6188	6190	6203	6223	6224	6225	6226	6229	6231	6232	6233	6240	6241	6242	6243
	6246	6256	6257	6258	6270	6274	6275	6276	6277	6282	6289	6290	6291	6292	6293
	6303	6305	6306	6307	6308	6309	6310	6315	6325	6326	6327	6331	6342	6349	6351
	6352	6353	6357	6358	6361	6364	6367	6372	6380	6381	6382	6383	6384	6385	6395
	6402	6404	6405	6406	6408	6410	6412	6414	6419	6429	6430	6431	6440	6442	6443
	6446	6454	6455	6456	6457	6458	6459	6464	6465	6466	6469	6486	6487	6497	6502
	6505	6514	6515	6516	6522	6525	6528	6536	6538	6540	6541	6548	6553	6554	6559
	6560	6561	6568	6571	6580	6585	6586	6591	6601	6614	6628	6630	6631	6632	6639

```
        6643  6646  6657  6663  6664  6669  6678  6686  6692  6698  6705  6712  6720  6722  6730
        6740  6741  6748  6759  6767  6769  6781  6788  6797  6799  6806  6814  6823  6845  6847
        6848  6854  6855  6856  6868  6869  6870  6875  6878  6885  6886  6887  6888  6889  6892
        6895  6899  6901  6906  6916  6917  6918  6919  6920  6921  6922  6928  6933  6934  6935
        6940  6941  6947  6948  6952  6953  6954  6955  6961  6966  6968  6969  6973  6974  6975
        6985  6991  7003  7005  7006  7012  7018  7019  7020  7029  7030  7031  7032  7037  7038
        7043  7052  7053  7060  7063  7067  7073  7074  7080  7087  7095  7097  7098  7105  7115
        7116  7117  7123  7128  7134  7142  7144  7156  7157  7163  7164  7172  7174  7175  7181
        7189  7198  7199  7211  7219  7225  7231  7232  7239  7240  7247  7256  7257  7261  7262
        7267  7268  7274  7280  7281  7282  7283  7288  7293  7294  7295  7300  7301  7302  7305
        7306  7307  7308  7312  7318  7319  7324  7333  7334  7335  7336  7341  7342  7345  7346
        7347  7352  7353  7356  7361  7367  7372  7378  7381  7382  7386  7396  7397  7398  7413
        7427  7441  7443  7448  7449  7450  7451  7455  7460  7464  7465  7472  7479  7494  7500
        7515  7518  7519  7530  7550  7556  7562  7569  7576  7583  7591  7597  7604  7610  7616
        7623
MOVB    5460  5478  5491  5492  5517  5521  5547  5572  5589  5610  5630  5648  5659  5684  5691
        5704  5728  6218  6219  6220  6235  6236  6237  6260  6261  6263  6265  6295  6296  6297
        6298  6332  6335  6338  6344  6387  6389  6391  6397  6432  6433  6436  6437  6460  6461
        6493  6498  6499  6517  6539  6542  6581  6607  6633  6647  6648  6658  6679  6699  6723
        6742  6753  6782  6789  6800  6824  7436  7444  7452  7468  7537
NEG     5881  5887  6136  7033  7039  7311
NEGB    5573  5579  6659  6665
NOP     5200  7503  7504  7505
ROL     5899  5900  5905  5907  7055  7056  7061  7064
ROLB    5591  5592  5597  5599  6681  6682  6687  6689
ROR     5921  5922  5926  5928  7076  7077  7081  7083
RORB    5612  5613  5617  5619  6701  6702  6706  6708
RTS     5374  5406  5414  7551  7557  7563  7570  7577  7584  7592  7598  7605  7611  7617  7624
RTT     7541
SBC     6015  6021  6023  6029  6032  6039  7177  7183  7185  7191  7194  7201
SBCB    5706  5712  5714  5720  5723  5730  6802  6808  6810  6816  6819  6826
SCC     5195  5212  5248  5266  5280  5296  5359  5405  5455  5472  5493  5548  5554  5561  5750
        5794  5856  5862  5869  6047  6069  6077  6091  6102  6488  6543  6634  6640  6649  6849
        6923  7007  7013  7021  7213  7233  7241  7258  7269  7373  7435  7442  7467
SEC     5159  5176  5518  5531  5676  5678  5685  5692  5711  5713  5719  5722  5729  5826  5839
        5985  5987  5994  6001  6020  6022  6028  6031  6038  6582  6596  6602  6774  6776  6783
        6790  6807  6809  6815  6818  6825  6970  6986  7149  7151  7158  7165  7182  7184  7190
        7193  7200
SEN     5121  5137  5157  5174
SEV     5138  5158  5175
SEZ     5177
SOB     7374  7383
SUB     6149  6156  6161  6167  7337  7343  7348  7354  7357
SWAB    6071  6079  7235  7243
SXT     6049  6055  7215  7221
TST     5087  5403  5753  5758  5813  6051  6272  6280  6313  6354  6359  6360  6362  6363  6367
        6366  6370  6386  6407  6417  6569  6590  6852  6858  7217  7315  7359  737E  7384  74[illegible]
        7466
TSTB    5458  5462  6268  6301  6347  6400  6491  6495  6589  7532  7535
XOR     6092  6098  6103  7259  7265  7270
.ABS    4893
.ASCIZ  5052  5054
.BLKW   5056
.BYTE   5031
.ENABL     4
.END    7626
```

.ENDC	4898	4908	4918	5031	5093	5116	5132	5152	5169	5190	5207	5227	5242	5251	5260
	5269	5290	5323	5353	5365	5383	5417	5449	5467	5485	5511	5541	5567	5584	5605
	5624	5643	5666	5698	5744	5769	5786	5819	5849	5875	5892	5914	5933	5952	5975
	6007	6041	6063	6084	6111	6141	6171	6213	6251	6284	6319	6374	6423	6448	6481
	6509	6532	6575	6624	6652	6673	6693	6716	6735	6763	6793	6841	6863	6911	6962
	6999	7024	7047	7068	7091	7110	7138	7168	7206	7226	7251	7275	7328	7362	7390
	7429														
.EVEN	5031	5053	5055												
.IF	4898	4908	4918	5031	5093	5116	5132	5152	5169	5190	5207	5227	5242	5251	5260
	5269	5290	5323	5353	5365	5383	5417	5449	5467	5485	5511	5541	5567	5584	5605
	5624	5643	5666	5698	5744	5769	5786	5819	5849	5875	5892	5914	5933	5952	5975
	6007	6041	6063	6084	6111	6141	6171	6213	6251	6284	6319	6374	6423	6448	6481
	6509	6532	6575	6624	6652	6673	6693	6716	6735	6763	6793	6841	6863	6911	6962
	6999	7024	7047	7068	7091	7110	7138	7168	7206	7226	7251	7275	7328	7362	7390
	7429														
.IFF	4908	4918													
.IIF	4898	5031	5093	5116	5132	5152	5169	5190	5207	5227	5242	5260	5290	5323	5353
	5417	5467	5485	5511	5541	5567	5584	5605	5624	5643	5666	5698	5769	5786	5819
	5849	5875	5892	5914	5933	5952	5975	6007	6041	6063	6084	6111	6141	6171	6251
	6284	6319	6374	6423	6448	6509	6532	6575	6624	6652	6673	6693	6716	6735	6763
	6793	6863	6911	6962	6999	7024	7047	7068	7091	7110	7138	7168	7206	7226	7251
	7275	7328	7362	7390	7429										
.LIST	2	4623	4896	4917	5031	5066	5093	5095	5112	5116	5118	5130	5132	5134	5148
	5152	5154	5167	5169	5171	5186	5190	5192	5206	5207	5209	5222	5227	5229	5231
	5234	5236	5238	5240	5242	5244	5250	5251	5254	5257	5260	5262	5268	5269	5272
	5275	5282	5285	5289	5290	5292	5298	5300	5303	5306	5312	5313	5316	5321	5323
	5325	5330	5333	5336	5339	5344	5348	5352	5353	5355	5361	5364	5365	5368	5371
	5375	5382	5383	5386	5389	5392	5393	5397	5401	5407	5412	5415	5417	5419	5440
	5443	5449	5451	5453	5467	5469	5477	5481	5484	5485	5487	5489	5498	5503	5511
	5513	5515	5530	5541	5543	5545	5553	5559	5567	5569	5577	5583	5584	5586	5596
	5602	5605	5607	5616	5623	5624	5626	5628	5637	5643	5645	5654	5665	5666	5668
	5670	5683	5690	5698	5700	5702	5710	5718	5727	5734	5744	5746	5748	5762	5769
	5771	5779	5782	5785	5786	5788	5799	5804	5815	5819	5821	5823	5838	5849	5851
	5853	5861	5867	5875	5877	5885	5891	5892	5894	5904	5910	5914	5916	5925	5932
	5933	5935	5937	5946	5952	5954	5963	5974	5975	5977	5979	5992	5999	6007	6009
	6011	6019	6027	6036	6041	6043	6053	6059	6063	6065	6075	6083	6084	6086	6107
	6111	6113	6121	6127	6135	6140	6141	6143	6145	6153	6158	6165	6171	6173	6179
	6180	6181	6183	6186	6188	6192	6193	6194	6196	6199	6203	6206	6213	6215	6223
	6229	6240	6246	6251	6253	6270	6282	6284	6286	6303	6315	6319	6321	6331	6342
	6349	6357	6372	6374	6376	6395	6402	6419	6423	6425	6440	6446	6448	6450	6464
	6469	6473	6481	6483	6502	6505	6509	6511	6522	6525	6528	6532	6534	6536	6548
	6553	6568	6571	6575	6577	6601	6614	6624	6626	6628	6639	6646	6652	6654	6663
	6669	6673	6675	6686	6692	6693	6695	6705	6712	6716	6718	6720	6730	6735	6737
	6748	6759	6763	6765	6767	6781	6788	6793	6795	6797	6806	6814	6823	6830	6841
	6843	6845	6863	6865	6875	6878	6895	6899	6906	6911	6913	6928	6933	6940	6952
	6961	6962	6964	6966	6985	6999	7001	7003	7012	7018	7024	7026	7037	7043	7047
	7049	7060	7067	7068	7070	7080	7087	7091	7093	7095	7105	7110	7112	7123	7134
	7138	7140	7142	7156	7163	7168	7170	7172	7181	7189	7198	7206	7208	7219	7225
	7226	7228	7239	7247	7251	7253	7274	7275	7277	7288	7293	7300	7318	7324	7328
	7330	7341	7345	7352	7361	7362	7364	7372	7378	7381	7386	7390	7393	7401	7402
	7403	7405	7408	7413	7416	7417	7418	7420	7423	7427	7429	7431	7441	7448	7455
	7460	7464	7472	7479	7482	7492	7494	7545	7550	7556	7562	7569	7576	7583	7591
	7597	7604	7610	7616	7623										
.MACRO	39	81	168	308	485	515	586	747	801	895	926	974	986	1030	1064
	1097	1110	1131	1144	1177	1226	1272	1309	1356	1389	1419	1475	1483	1535	1739
	1947	2140	2228	2353	2450	2528	2613	2827	2919	2995	3095	3223	3286	3348	3466

```
                3504    3568    3666    3751    3790    3853    3891    3934    4032    4080    4350    4397    4437    4513    4920
                4933    4948    4961    4969    4976    4982    4986    4998    5001    5005
.MCALL          4897
.MEXIT          5031
.NLIST             1       3    4895    4912    5031    5064    5093    5095    5112    5116    5118    5130    5132    5134    5148
                5152    5154    5167    5169    5171    5186    5190    5192    5206    5207    5209    5222    5227    5229    5231
                5234    5236    5238    5240    5242    5244    5250    5251    5254    5257    5260    5262    5268    5269    5272
                5275    5282    5285    5289    5290    5292    5298    5300    5303    5306    5312    5313    5316    5321    5323
                5325    5330    5333    5336    5339    5344    5348    5352    5353    5355    5361    5364    5365    5368    5371
                5375    5382    5383    5386    5389    5392    5393    5397    5401    5407    5412    5415    5417    5419    5440
                5441    5449    5451    5453    5467    5469    5477    5481    5484    5485    5487    5489    5498    5503    5511
                5513    5515    5530    5541    5543    5545    5553    5559    5567    5569    5577    5583    5584    5586    5596
                5602    5605    5607    5616    5623    5624    5626    5628    5637    5643    5645    5654    5665    5666    5668
                5670    5683    5690    5698    5700    5702    5710    5718    5727    5732    5744    5746    5748    5762    5769
                5771    5779    5782    5785    5786    5788    5799    5804    5815    5819    5821    5823    5838    5849    5851
                5853    5861    5867    5875    5877    5885    5891    5892    5894    5904    5910    5914    5916    5925    5932
                5933    5935    5937    5946    5952    5954    5963    5974    5975    5977    5979    5992    5999    6007    6009
                6011    6019    6027    6036    6041    6043    6053    6059    6063    6065    6075    6083    6084    6086    6107
                6111    6113    6121    6127    6135    6140    6141    6143    6145    6153    6158    6165    6171    6173    6178
                6181    6183    6184    6185    6188    6191    6194    6196    6197    6198    6203    6204    6213    6215    6223
                6229    6240    6246    6251    6253    6270    6282    6284    6286    6303    6315    6319    6321    6331    6342
                6349    6357    6372    6374    6376    6395    6402    6419    6423    6425    6440    6446    6448    6450    6464
                6469    6471    6481    6483    6502    6505    6509    6511    6522    6525    6528    6532    6534    6536    6548
                6553    6568    6571    6575    6577    6601    6614    6624    6626    6628    6639    6646    6652    6654    6663
                6669    6673    6675    6686    6692    6693    6695    6705    6712    6716    6718    6720    6730    6735    6737
                6748    6759    6763    6765    6767    6781    6788    6793    6795    6797    6806    6814    6823    6828    6841
                6843    6845    6863    6865    6875    6878    6895    6899    6906    6911    6913    6928    6933    6940    6952
                6961    6962    6964    6966    6985    6999    7001    7003    7012    7018    7024    7026    7037    7043    7047
                7049    7060    7067    7068    7070    7080    7087    7091    7093    7095    7105    7110    7112    7123    7134
                7138    7140    7142    7156    7163    7168    7170    7172    7181    7189    7198    7206    7208    7219    7225
                7226    7228    7239    7247    7251    7253    7274    7275    7277    7288    7293    7300    7318    7324    7328
                7330    7341    7345    7352    7361    7362    7364    7372    7378    7381    7386    7390    7393    7400    7403
                7405    7406    7407    7413    7415    7418    7420    7421    7422    7427    7429    7431    7441    7448    7455
                7460    7464    7472    7479    7480    7492    7494    7543    7550    7556    7562    7569    7576    7583    7591
                7597    7604    7610    7616    7623
.NTYPE          6181    6183    6194    6196    7403    7405    7418    7420
.PAGE           4887    4889    4891    4905    5030    5033    5067    5093    5132    5169    5207    5242    5290    5353    5444
                5485    5541    5584    5624    5666    5735    5786    5849    5892    5933    5975    6041    6084    6141    6207
                6284    6374    6448    6474    6509    6575    6652    6693    6735    6793    6831    6863    6962    7024    7068
                7110    7168    7226    7275    7362    7429    7483    7522    7546
.REPT           4624    4913
.SBTTL          4918    5031    5032    5065    5093    5116    5132    5152    5169    5190    5207    5227    5242    5260    5290
                5323    5353    5417    5442    5449    5467    5485    5511    5541    5567    5584    5605    5624    5643    5666
                5698    5733    5744    5769    5786    5819    5849    5875    5892    5914    5933    5952    5975    6007    6041
                6063    6084    6111    6141    6171    6205    6213    6251    6284    6319    6374    6423    6448    6472    6481
                6509    6532    6575    6624    6652    6673    6693    6716    6735    6763    6793    6829    6841    6863    6911
                6962    6999    7024    7047    7068    7091    7110    7138    7168    7206    7226    7251    7275    7328    7362
                7390    7429    7481    7512    7525    7544
.TITLE          4898
.WORD           4918    5031    5032    6181    6183    6194    6196    7403    7405    7418    7420
```

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

L09


```
*,DVKAAA/DS:ERFZ/CRF=DVKAAA.SML,DVKAAA.P11
RUN-TIME: 37 49 8 SECONDS
RUN-TIME RATIO: 1533/95=16.0
CORE USED:  30K  (59 PAGES)
```

Spooler runtime 14 Seconds, 61 KCS, 406 disk reads, 4 disk writes, 115 pages