MAIN DEC CHANGE NOTICE
MAY BE REQUIRED FOR
PROGRAM TO OPERATE

IDENTIFICATION

PRODUCT CODE	MAINDEC-11-DZQKC-E-D
PRODUCT NAME	11 FAMILY INSTRUCTION EXERCISER
DATE CREATED	MARCH 21, 1975
MAINTAINER	DIAGNOSTIC GROUP
AUTHOR	J. ADAMS

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

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

1.0 ABSTRACT

THIS DIAGNOSTIC PROGRAM IS DESIGNED TO BE A COMPREHENSIVE CHECK OF THE PDP11/05 AND PDP11/20 PROCESSORS. THE PROGRAM EXECUTES EACH INSTRUCTION IN ALL ADDRESS MODES AND INCLUDES TESTS FOR TRAPS AND THE TELETYPE INTERRUPT SEQUENCE. THE PROGRAM DOES NOT TEST INSTRUCTIONS NOT COMMON TO THE 11/20 OR 11/05. THE PROGRAM RELOCATES THE TEST CODE THROUGHOUT MEMORY 0-28K.

2.0 REQUIREMENTS

2.1 EQUIPMENT

PDP11 FAMILY CENTRAL PROCESSOR
OPTIONAL - KW11-L (LINE CLOCK)

2.2 STORAGE

THE PROGRAM USES ALL OF THE FIRST 4K OF MEMORY (EXCLUDING THAT AREA OF MEMORY RESERVED FOR THE LOADERS).

2.3 PRELIMINARY PROGRAMS

NONE, HOWEVER, THE EMT AND TRAP INSTRUCTION SHOULD BE VERIFIED BEFORE RUNNING.

3.0 LOADING AND STARTING PROCEDURE

LOAD PROGRAM USING ABS LJADER
IF THE CONSOLE TTY IS A SERIAL LA30 OR A VT05 FILLER CHARACTERS MAY BE REQJIRED. DEPOSIT INTO LOCATION 1002 (FILLS) A 4400.
LOAD ADDRESS = 200
PRESS START
SET OPERATING SWITCHES
PASS COUNT IS PRINTED AFTER EACH PASS (SEE SEC 6.4)
"DZQKC DONE" IS PRINTED WHEN DONE (SEE SEC 7.1)

4.0 SWITCH SETTINGS

SW15 HALT ON ERROR... THIS SWITCH WHEN SET WILL HALT THE PROCESSOR WHEN AN ERROR IS DETECTED. THE PC+2 AND THE CURRENT STATUS AT THE TIME OF THE ERROR IS STORED ON THE STACK (R6). IF THIS SWITCH IS SET BEFORE AN ERROR IS DETECTED THE PROGRAM HALTS AS DESCRIBED AROVE. THE PROGRAM MAY BE HALTED AFTER THE ERROR TYPEOUT OCCURS BY SETTING SW15 AFTER THE TYPEOUT BEGINS.

SW14 LOOP SUBTEST... THIS SWITCH WHEN SET LOOPS THE CURRENT SUBTEST RUNNING REGARDLESS OF ERROR.

SW13 INHIBIT ERROR PRINTOUT - THIS SWITCH WHEN SET INHIBITS THE ERROR PRINTOUT.

SW12 INHIBIT RELOCATION... THIS SWITCH WHEN SET CAUSES THE PROGRAM TO BE EXECJTED ONLY IN THE FIRST 4K OF MEMORY. THIS SWITCH CANNOT BE SET WHEN THE PROGRAM IS RUNNING.

SW11 INHIBIT SUBTEST ITERATION... THIS SWITCH WHEN SET INHIBITS SUBTEST REITERATION. NORMALLY EACH SUBTEST IS EXECUTED 8 TIMES BEFORE THE NEXT SUBTEST IS RUN.

SETTING SW11 CAUSES EACH TEST TO BE EXECUTED ONCE BEFORE STARTING THE NEXT SUBTEST.

SW1Ø RING BELL ON ERROR... THIS SWITCH WHEN SET WILL RING THE BELL WHEN AN ERROR IS DETECTED.

SW7 INHIBIT ALL BUT ERROR TYPEOUTS...THIS SWITCH WHEN RESET (Ø) INHIBITS THE END OF PASS TYPEOUT (ICNT=XXXX) AND THE END OF PROGRAM TYPEOUT (DZQKC DONE)

5.Ø ERRORS

IF AN ERROR IS DETECTED THE PROGRAM WILL TRAP TO THE ERROR HANDLING ROUTINE (ERROR). IF ENABLED THIS ROUTINE WILL BYTE THE PC AND THE PROCESSER STATUS AT THE TIME OF THE ERROR. ALSO (IF REQUIRED) THE ORIGINAL PC (WHERE THE PC WAS RELOCATED FROM).

5.Ø.1 ERROR PRINTOUT FORMAT

```
ICNT=AAAA PC=BBBBBB PSW=DDDDDD
OR
ICNT=AAAA PC=BBBBBB PSW=DDDDDD PC RELOCATED FROM CCCCCC

WHERE:  AAAA=PASS COUNT
        BBBBBB=PC AT THE TIME OF THE ERROR
        CCCCCC=PC OF THE ORIGINAL CODE RELOCATED
        DDDDDD=PSW AT THE TIME OF THE ERROR.
```

5.1 PARITY ERROR DECTECTION

IF A PARITY ERROR IS DETECTED THE PROGRAM WILL TYPE A MESSAGE"PARITY ERROR" AND SCAN MEMORY FOR THE PARITY ERROR. WHEN THE FAILING ADDRESS IS LOCATED THE PROGRAM WILL HALT WITH THE VALUE OF THE ADDRESS+2 IN RØ.

5.2 ERROR LOOPING

THE SUBTEST DETECTING THE ERROR MAY BE LOOPED INDEFINITELY BY SETTING SW14. SETTING SW13 WILL INHIBIT THE TYPEOUT AND ALLOW SCOPING THE FAULTY SIGNAL(S).

5.3 UNPREDICTED ERRORS

THE PROGRAM MAY ON OCCASSION DETECT A MEMORY ERROR THE RESULTS OF WHICH WERE NOT PREDICTABLE IN WHICH CASE THE PROGRAM MAY BEHAVE UNPREDICTABLY. WHEN THIS HAPPENS THE USER MUST RETRACE THE PROGRAM STEPS TO RESOLVE WHERE THE ERROR OCCURRED. THE FOLLOWING ITEMS SHOULD BE CONSIDERED AND MAY BE OF USE WHEN RETRACING A FAILURE OF THIS NATURE.

1. HALT THE PROGRAM (IF NECESSARY)

2. EXAMINE RELR1

   ADDRESS RELR1 (1ØØ6) CONTAINS THE UNRELOCATED VALUE OF THE PC OF THE LAST TEST THAT WAS SUCCESSFULLY EXECUTED.

3. EXAMINE FACTOR
   ADDRESS FACTOR (1ØØ4) CONTAINS THE RELOCATION FACTOR.

4. EXAMINE ALL LOCATIONS STARTING WITH THE ADDRESS SPECIFIED IN R1/R11(IF PSW BIT11 =Ø/1) COMPARING THEIR CONTENTS WITH THE CONTENTS OF THE CORRESPONDING UNRELOCATED CODE (SPECIFIED IN 1ØØ6) AS SHOWN IN THE LISTING. EXAMINE AND COMPARE UNTIL EITHER A DIFFERENCE IN INSTRUCTION (I.E., THE ERROR) OR THE NEXT 'SCOPE' IS SEEN.

IF THE PROGRAM TRAPS AND HALTS AT A TRAP/INTERRUPT VECTOR+2 (NOTE: THE PDP-11/45 WILL DISPLAY THE ADDRESS OF THE HALT+2 I.E., A FALSE TRAP TO 4 WILL DISPLAY 1Ø).

1A. EXAMINE THE STACK (R6)

THE TOP WORD ON THE STACK CONTAINS THE PC AT THE TIME OF THE TRAP. IF THE PC IS GREATER THAN 2ØØØØ, THEN

2A. EXAMINE LOCATION 1ØØ2 (FACTOR)

THIS LOCATION CONTAINS THE PROGRAM RELOCATION FACTOR WHICH, WHEN SUBTRACTED FROM THE PC GIVES THE PC OF THE ORIGINAL CODE.

6.Ø SUBROUTINE ABSTRACTS

6.1 SCOPEA

THE SCOPEA ROUTINE IS ENTERED BY THE SCOPE (EMT) INSTRUCTION AND IS EXECUTED AT THE START OF EACH SUBTEST. THE ROUTINE MONITORS SW14, SW11 AND SW 8 AND TAKES APPROPRIATE ACTION. ALSO, THIS ROUTINE STORES IN R1/R11 THE FIRST ADDRESS OF THE SUBTEST BEING ENTERED.

6.2 ERROR

THE ERROR ROUTINE IS ENTERED BY THE HLT (TRAP) INSTRUCTION, AND IS EXECUTED WHEN A PREDICTABLE ERROR IS DETECTED. THIS ROUTINE MONITORS SW15, SW13, AND SW1Ø.

6.3 RELOC

THE RELOC ROUTINE IS ENTERED BY A MOV RELOC,PC INSTRUCTION. THIS ROUTINE RELOCATES THE PROGRAM CODE THROUGHOUT MEMORY, AND 'JUMPS' TO THE RELOCATED CODE AFTER IT HAS BEEN MOVED SUCCESSFULLY. IF THE CODE CANNOT BE RELOCATED (BECUASE OF INSUFFICIENT MEMORY) THE ROUTINE 'JUMPS' TO THE NEXT SECTION OF UNRELOCATED PROGRAM CODE. THE CODE MOVED IS LESS THAN 1K (4ØØØ) BYTES). AT THE START AND END OF EACH SECTION OF CODE TO BE MOVED ARE A SECTION OF CODE WHICH ESTABLISHES THE FIRST ADDRESS OF THE CODE TO BE MOVED, AND SETS A SCOPE POINTER (R1/R11) AND, ALSO A SECTION WHICH ESTABLISHES THE LAST ADDRESS AND 'JUMPS' TO THE RELOCATION (RELOC) ROUTINE. EACH SECTION OF CODE IS

IDENTIFIED AS SHOWN BELOW!

```
;0000000000000FIRST ADDRESS TO BE RELOCATED00000000

     CODE TO BE MOVED AND EXECUTED

;0000000000000LAST ADDRESS OF CODE TO BE RELOCATED 00000000
```

THE RELOC ROUTINE DOES NOT RELOCATE PROGRAM CODE INTO THE LAST 1000(8) BYTES OF MEMORY, THUS PRESERVING THE LOADERS.

6.4 END

THIS ROUTINE IS ENTERED AT THE COMPLETION OF EACH PASS IT SETS UP (LOADS NEW PROCESSOR STATUS) FOR THE NEXT PASS; AND PRINTS THE PASS COUNT!

```
     ICNT=XXXX
```

7.0 MISCELLANEOUS

7.1 EXECUTION TIME

THE EXECUTION TIME IS HIGHLY VARIABLE (DEPENDENT ON PROCESSOR, TYPE OF MEMORY, AND AMOUNT OF MEMORY). HOWEVER, WHEN THE PROGRAM IS RUNNING SUCCESSFULLY THERE IS A NOTICEABLE 'FLICKER' DISPLAYED IN THE CONSOLE LIGHT PATTERN THE 'FLICKER' WILL DIM WHEN 'T' BIT TRAP PASSES (EVERY ODD PASS) ARE RUNNING, THE PROGRAM SHOULD BE RUN FOR A MINIMUM OF:

          2 PASSES ICNT=2 11/05 OR 11/20

SOME TYPICAL TIMES FOLLOW!

8.0 PROGRAM DESCRIPTION

THE PROGRAM IS DIVIDED INTO FOUR SECTIONS OF POSITION INDEPENDENT RELOCATABLE TEST CODE. EACH SECTION IS APPROXIMATELY 1K WORDS LONG. (EXCEPT SECTION A).

SECTION 0 THIS SECTION TEST THE UNARY INSTRUCTION SET EXECUTING EACH UNARY INSTRUCTION IN EACH ADDRESS MODE (EXCLUDING UNARY INSTRUCTIONS USING ADDRESS MODE 7).

SECTION 1 THIS SECTION TESTS THE UNARY INSTRUCTIONS USING ADDRESS MODE 7 AND BINARYS IN ALL ADDRESS MODES (EXCLUDING BINARY BYTE OPS USING ADDRESS MODE 7).

SECTION 2 THIS SECTION TEST BINARY BYTE OPS USING ADDRESS MODE 7, JMP, JSR AND PROGRAM TRAP (IOT, TRAP AND EMT) INSTRUCTIONS,

SECTION A FOLLOWING SECTION 2 IS A ROUTINE TO ASCERTAIN WHICH CP THE PROGRAM IS RUNNING ON. THE RESULTS ARE USED BY THE FOLLOWING CODE TO CHECK THE ADDITIONAL INSTRUCTIONS/FEATURES OF THE 11/40 AND 11/45.

SECTION 3 THIS SECTION CHECKS THAT EACH BIT IN THE PROCESSOR STATUS WORD (PSW) CAN BE SET CLEARED, RESERVED INSTRUCTION, AND ODD ADDRESS TRAPS.

FOLLOWING SECTION 3 ARE TWO ROUTINES TO CHECK THE TELETYPE PRINTER LOGIC AND A ROUTINE TO START THE KW11-L LINE CLOCK. IF THE KW11-L IS AVAILABLE THE PRIORITY ARBITRATION LOGIC IS TESTED.

AFTER EACH INDIVIDUAL SECTION HAS BEEN EXECUTED THE "RELOC" ROUTINE WILL RELOCATE THE SECTION THROUGHOUT ALL MEMORY UP TO 28K. WHEN THE SECTION HAS BEEN RELOCATED AND EXECUTED IN ALL MEMORY THE "RELOC" ROUTINE WILL RETURN THE PROGRAM TO THE NEXT UNRELOCATED SECTION.

RELOCATION AND EXECUTION OF ALL SECTIONS THROUGHOUT ALL MEMORY CONSTITUTES A SINGLE PASS.

UPON COMPLETION OF A PASS OF THE PROGRAM THE PROGRAM RESTARTS USING A NEW PROCESSOR STATUS DEPENDING ON THE TYPE OF PROCESSOR AND THE PASS COUNT.

8.1 STACK POINTER

THE STACK POINTER IS SET AT 500.
NOTE: IF THE PROGRAM IS RUNNING IN EITHER USER OR SUPERVISOR MODE (NOT APPLICABLE IF 11/20 OR 11/05) THE USER/SUPERVISOR STACK POINTER IS SET TO 500 AND THE KERNEL STACK POINTER IS SET TO 600. THE KERNEL STACK POINTER IS USED ONLY FOR THE SCOPE, HLT, TTY, AND KW11-L (IF AVAILABLE TRAP/INTERUPT ROUTINES.

8.2 POWER FAILURE

A POWER FAIL SERVICE ROUTINE IS INCORPORATED IN THE TEST. WHEN USING THIS PROGRAM THE POWER SHOULD BE TURNED OFF WHEN RUNNING TO CHECK THE POWER FAIL LOGIC. WHEN THE POWER FAILS THE PROGRAM WILL TYPE:

POWER FAILED

AND RESTART THE PROGRAM AT THE BEGINNING. (START)

9.0 USER DEFINED RELOCATION LIMITS

THE PROGRAM WILL REQUEST A LOWER AND UPPER LIMIT FOR RELOCATION. THE LIMITS MUST BE BETWEEN 20000 AND 157776. THE PROGRAM WILL EXECUTE IN THE LOWER 4K (0-17776) AND THE LIMITS SPECIFIED.
THE STARTING ADDRESS IS 204.
TO RETAIN PREVIOUSLY SPECIFIED LIMITS START AT 210.

```
                                        .NLIST  SEQ?MD,MC
                                        .LIST   ME
                                        .ABS
                                        .TITLE  FRONT END
                                ;CONTAINS DEFINITIONS, REGISTER ASSIGNMENTS AND MACRO CALLS

                                ;GENERAL REGISTER ASSIGNMENTS
        000000                          R0=%0
        000001                          R1=%1
        000002                          R2=%2
        000003                          R3=%3
        000004                          R4=%4
        000005                          R5=%5
        000006                          SP=%6
        000007                          PC=%7
        000000                          R10=%0
        000001                          R11=%1
        000002                          R12=%2
        000003                          R13=%3
        000004                          R14=%4
        000005                          R15=%5


                                ;STATUS REGISTER (PSW) BIT ASSIGNMENTS
        000001                          C=1                     ;C BIT
        000002                          V=2                     ;V BIT
        000004                          Z=4                     ;Z BIT
        000010                          N=10                    ;N BIT
        000020                          T=20                    ;'T' BIT
        000340                          PRTY7=340               ;PRIORITY LEVEL 7
        000300                          PRTY6=300               ;PRIORITY LEVEL 6
        000200                          PRTY4=200               ;PRIORITY LEVEL 4

                                ;VECTOR ADDRESSES
        000004                          ERRVEC=4                ;ADDRESS OF ERROR VECTOR
        000010                          RESVEC=10               ;ADDRESS OF RESERVED INST. TRAP VECTOR
        000014                          TBITVEC=14              ;ADDRESS OF 'T' BIT TRAP VECTOR
        000014                          TRTVEC=14               ;ADDRESS OF 'TRACE' TRAP VECTOR
        000014                          BPTVEC=14               ;ADDRESS OF 'BREAKPOINT' TRAP VECTOR
        000020                          IOTVEC=20               ;ADDRESS OF IOT TRAP VECTOR
        000024                          PFVEC=24                ;ADDRESS OF POWER FAIL TRAP VECTOR
        000030                          EMTVEC=30               ;ADDRESS OF EMT VECTOR
        000034                          TRAPVEC=34              ;ADDRESS OF TRAP VECTOR
        000064                          TPVEC=64                ;ADDRESS OF TTY PRINTER INTERRUPT VECTOR
        000100                          LKVEC=100               ;ADDRESS KW11-L LINE CLOCK INT. VECTOR
        000240                          PIRVEC=240              ;ADDRESS OF PIRQ VECTOR
        000244                          FPEVEC=244              ;ADDRESS OF FLOATING POINT INT. VECTOR
        000250                          MMVEC=250               ;ADDRESS OF MEM MGMT ERROR TRAP VECTOR

                                ;REGISTER ADDRESSES
        177776                          PSW=    177776          ;ADDRESS OF STATUS REGISTER
        177774                          SLR=    177774          ;ADDRESS OF STACK LIMIT REGISTER
        177772                          PIRQ=   177772          ;ADDRESS OF PROGRAM INTERRUPT REQUEST
```

```
        177770                          UBREAK= 177770          ;ADDRESS OF MICRO BREAK REGISTER
        177546                          LKS=    177546          ;ADDRESS OF KW11-L STATUS REG.
        177560                          TKS=    177560          ;ADDRESS OF KEYBOARD CSR
        177562                          TKB=    177562          ;ADDRESS OF KEYBOARD BUFFER
        177564                          TPS=    177564          ;ADDRESS OF TELEPRINTER CSR
        177566                          TPB=    177566          ;ADDRESS OF TELEPRINTER BUFFER
        177572                          SR0=    177572          ;ADDRESS OF MEM MGMT REGISTER SR0
        177570                          SWR=    177570          ;ADDRESS OF CONSOL SWITCH REGISTER
        177570                          DISPLAY=177570          ;ADDRESS OF CONSOL DISPLAY REGISTER
        177514                          LPS=    177514          ;ADDRESS OF LINE PRINTER STATUS REG
        177516                          LPB=    177516          ;ADDRESS OF LINE PRINTER DATA DUFFER

                                ;INITIAL STACK POINTER SETTING
        000500                          STKPTR= 500             ;PROGRAM STACK PTR
        000600                          KPTR=600                ;KERNEL STACK PTR (USED BY KERNEL WHEN
                                                                ;PROGRAM IS RUNNING IN OTHER THAN KERNEL
                                                                ;MODE (NOT APPLICABLE TO 11/05,11/20)

                                ;MISCELLANEOUS BIT ASSIGNMENTS
        100000                          BIT15=100000
        040000                          BIT14=40000
        020000                          BIT13=20000
        000400                          BIT8=400
        000100                          BIT6=100

                                ;INSTRUCTION EQUATES
        104400                          HLT=TRAP                ;HLT IS A TRAP INST TO THE ERROR ROUTINE
        104000                          SCOPE=EMT               ;SCOPE IS AN EMT TRAP




        000046                          .=46
000046  016542                          LOGICAL
        000052                          .=52
000052  040000                          BIT14
        000200                          .=200
000200  012707  002066                  MOV     #START,PC       ;GO TO START OF TEST
000204  012707  002160                  MOV     #START1,PC      ;GO GET LOWER/UPPER RELOCATION BOUNDARY
000210  012707  002224                  MOV     #START3,PC      ;START WITH LAST TYPED BOUNDARY LIMITS

                                ;ROUTINE TO SAVE REGISTERS ON THE STACK
```

```
                                         ;CALLED BY SAVE MACRO OR JSR    PC,$SAVR
        000214  012667  000016           $SAVR:  MOV     (SP)+,1$        ;SAVE RETURN PC
        000220  010546                           MOV     %5,-(SP)
        000222  010446                           MOV     %4,-(SP)
        000224  010346                           MOV     %3,-(SP)
        000226  010246                           MOV     %2,-(SP)
        000230  010146                           MOV     %1,-(SP)
        000232  010046                           MOV     %0,-(SP)
        000234  012707                           MOV     (PC)+,PC        ;RETURN
        000236  000000                   1$:     0                       ;CONTAINS RETURN ADDRESS

                                         ;ROUTINE TO RESTORE REGISTERS SAVED ON THE STACK
                                         ;CALLED BY RESTORE MACRO OR JSR  PC,$RESTR
        000240  012667  000016           $RESTR: MOV     (SP)+,1$        ;SAVE RETURN PC
        000244  012600                           MOV     (SP)+,%0
        000246  012601                           MOV     (SP)+,%1
        000250  012602                           MOV     (SP)+,%2
        000252  012603                           MOV     (SP)+,%3
        000254  012604                           MOV     (SP)+,%4
        000256  012605                           MOV     (SP)+,%5
        000260  012707                           MOV     (PC)+,PC        ;RETURN
        000262  000000                   1$:     0                       ;CONTAINS RETURN ADDRESS

                000610                           .=610
                                         ;POWER FAIL SUBROUTINE
        000610  012737  000620  000024   PDWN:   MOV     #PUP,@#PFVEC
        000616  000000                           HALT

                                         ;POWER UP SUBROUTINE
        000620  012737  000610  000024   PUP:    MOV     #PDWN,@#PFVEC   ;RESTORE POWER FAIL TRAP TO POWER
                                                                         ;DOWN ROUINE ABOVE
        000626  012706  000600                   MOV     #KPTR,SR        ;SET STACK PTR
        000632  005027                           CLR     (PC)+
        000634  000000                   1$:     .WORD   0               ;KILL TIME
        000636  005267  177772           2$:     INC     1$
        000642  001375                           BNE     2$
        000644  004767  000362                   JSR     PC,,PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        000650  000656                           PFAIL
        000652  000137  002066                   JMP     @#START                 ;RESTART TEST

        000656  005015  047520  042527   PFAIL:  .ASCIZ  <15><12>'POWER FAILED'<15><12>
        000664  020122  040506  046111
        000672  042105  005015     000

                000740                           .=740
                                         ;NOTE: THIS CODE USED ONLY BY THE XOR TESTER.
                                         ;TO USE CODE PLACE 776 (BR  .-2) IN SCOPEA
        000740  012737  000002  000006   FORXOR: MOV     #RTI,@#ERRVEC+2 ;SET TIME OUT TRAP TO RETURN
        000746  000261                           SEC                     ;SET C
        000750  005737  177060                   TST     @#177060        ;IF A TIME OUT OCCURS THEN WHEN NEXT
                                                                         ;INSTRUCTION IS EXECUTED 'C' WILL BE SET
                                                                         ;AND IF NO TIME OUT 'C' WILL BE CLEARED
        000754  103401                           BCS     1$              ;BRANCH IF 'C' SET (TIMED OUT)
        000756  011601                           MOV     (SP),R1         ;ADDRESS OF NEXT SUBTEST TO R1
```

```
        000760  005037  000006           1$:     CLR     @#ERRVEC+2      ;RESTORE TIME OUT TRAP
        000764  010116                           MOV     R1,(SP)         ;GET RETURN ADDRESS BACK TO SUBTESTS
        000766  000240                           NOP
        000770  000002                           RTI                     ;RETURN EITHER TO LAST OR NEXT SUBTEST

                000776                           .=776
        000776  000000                   TICKS:  .WORD   0               ;CONTAINS CLOCK TICK COUNT
                001000                           .=1000
        001000  000000                   ICNT:   0                       ;CONTAINS PASS COUNT
        001002  000000                   $FILLS: .WORD   0               ;CONTAINS FILLS COUNT IN ODD BYTE
                                                                         ;AND FILLER CHARACTER IN EVEN BYTE
        001004  000000                   FACTOR: 0 -                     ;CONTAINS RELOCATION FACTOR
                                         ;SUBTRACT # IN FACTOR FROM PC TO GET PC OF ORIGINAL CODE
        001006  000000                   RELR1:  0                       ;CONTAINS RELOCATED R1 (THE R1 OF THE
                                                                         ;ORIGINAL CODE MOVED)
        001010  000000                   FRSTAD: .WORD   0               ;CONTAINS FIRST ADRS OF CODE TO BE MOVED
        001012  000000                   FRSTMEM:.WORD   0               ;CONTAINS LOWER RELOCATION BOUNDARY ADDRESS
        001014  000751                           BR      FORXOR          ;BRANCH TO XOR TESTER CODE
                                         ;SCOPE (EMT) SERVICE ROUTINE
                                         ;THIS ROUTINE ALLOWS THE SUBTEST TO BE CONTINUOUSLY LOOPED, ITERATED
                                         ;(OR NOT ITERATED) BEFORE BEGINNING NEXT SUBTEST
        001016  000240                   SCOPEA: NOP
        001020  032766  004000  000002           BIT     #4000,2(SP)     ;WAS REGISTER SET BIT SET ON TRAP
        001026  001403                           BEQ     2$              ;BRANCH IF NOT
        001030  052737  004000  177776           BIS     #4000,@#PSW     ;RETAIN REGISTER SET
        001036  032737  040000  177570   2$:     BIT     #40000,@#SWR    ;CHECK BIT 14 (CONTINUOUS LOOP)
        001044  001416                           BEQ     SCOPEC
        001046  010116                   SCOPEB: MOV     R1,(SP)         ;LOAD RETURN ADDRESS
        001050  010137  001006                   MOV     R1,@#RELR1
        001054  163737  001004  001006           SUB     @#FACTOR,@#RELR1        ;RELR1 CONTAINS UNRELOCATED R1
        001062  032737  000400  177570           BIT     #400,@#SWR      ;LOAD PDP11/45 MICRO BREAK REG?
        001070  001403                           BEQ     1$
        001072  113737  177570  177770           MOVB    @#SWR,@#UBREAK  ;LOAD MICRO BREAK REG WITH SR0-7
        001100  000002                   1$:     RTI                     ;RETURN TO SUBTEST
        001102  032737  004000  177570   SCOPEC: BIT     #4000,@#SWR     ;SUBTEST ITERATION DESIRED?
        001110  001006                           BNE     SCOREE          ;BRANCH IF NO ITERATION DESIRED?
        001112  005327                           DEC     (PC)+           ;DECREMENT SUBTEST ITERATION COUNT
        001114  000040                   SCOPED: 40                      ;CONTAINS SUBTEST ITERATION COUNT
        001116  001353                           BNE     SCOPEB
        001120  012767  000040  177766   SCOPEF: MOV     #40,SCOPED      ;RESET ITERATION COUNT
        001126  011601                   SCOPEE: MOV     (SP),R1         ;GET ADDRESS OF NEXT TEST
        001130  000746                           BR      SCOPEB

                                         ;ROUTINE TO RELOCATE PROGRAM CODE
        001132  032737  010000  177570   RELOC:  BIT     #10000,@#SWR    ;CHECK IF RELOCATION DESIRED (BIT12)
        001140  001031                           BNE     3$              ;BRANCH IF NO RELOCATION DESIRED
        001142  013700  001010                   MOV     @#FRSTAD,R0     ;GET FIRST ADDRESS OF CODE TO BE MOVED
        001146  010005                           MOV     R0,R5           ;SAVE
        001150  010204                           MOV     R2,R4           ;GET LAST ADDRESS OF CODE TO BE MOVED
        001152  160504                           SUB     R5,R4           ;R4 CONTAINS # OF WORDS TO RELOCATE
        001154  010203                           MOV     R2,R3           ;SAVE LAST ADDRESS OF CODE TO BE MOVED
        001156  005737  001004                   TST     @#FACTOR        ;FIRST RELOCATION IS TO 20000
        001162  001004                           BNE     10$
        001164  010237  001230                   MOV     R2,@#RETPC      ;SAVE RETURN PC TO NEXT SECTION OF CODE
        001170  013702  001012                          MOV     @#FRSTMEM,R2    ;SET FIRST ADDDRESS
        001174  060204                          10$:    ADD     R2,R4           ;R4 CONTAINS LAST MEMORY ADDRESS
                                                                                ;TO BE USED
        001176  020437  002140                          CMP     R4,@#LSTMEM     ;CHECK IF SUFFICIENT MEMORY REMAINS
        001202  101011                                  BHI     4$
        001204  012022                          1$:     MOV     (R0)+,(R2)+     ;RELOCATE PROGRAM CODE
        001206  020003                                  CMP     R0,R3           ;CHECK IF DONE
        001210  001375                                  BNE     1$
        001212  024042                          2$:     CMP     -(R0),-(R2)     ;CHECK THAT CODE WAS RELOCATED
        001214  001401                                  BEQ     .+4             ;PROPERLY
        001216  104400                                  HLT                     ;ERROR! CODE NOT RELOCATED PROPERLY
        001220  020005                                  CMP     R0,R5           ;CHECK IF FINISHED CHECKING
        001222  001373                                  BNE     2$
        001224  010207                          3$:     MOV     R2,RC           ;GO EXECUTE RELOCATED CODE
        001226  011707                          4$:     MOV     (PC),PC         ;RETURN TO NEXT SECTION OF CODE
        001230  000000                          RETPC:  0                       ;CONTAINS PC OF NEXT SECTION OF CODE

                                                ;ROUTINE TO PRINT ASCII MESSAGE, MESSAGE MUST TERMINATE WITH A 0 BYTE,
        001232  010046                          .PRINT: MOV     R0,-(SP)        ;SAVE R0 ON THE STACK
        001234  017600  000002                          MOV     @2(SP),R0       ;GET MESSAGE ADDRESS
        001240  062766  000002  000002                  ADD     #2,2(SP)        ;ADJUST RETURN PC

        001246  112046                          1$:     MOVB    (R0)+,-(SP)     ;PUSH CHAR ON THE STACK
        001250  001003                                  BNE     2$              ;BRANCH IF NOT TERMINATOR
        001252  005726                                  TST     (SP)+           ;POP TERMINATOR OFF THE STACK
        001254  012600                                  MOV     (SP)+,R0        ;RESTORE R0
        001256  000207                                  RTS     PC              ;RETURN

        001260  004767  000026                  2$:     JSR     PC,5$           ;TYPE CHARACTER
        001264  122726  000012                  3$:     CMPB    #12,(SP)+       ;CHECK IF CHAR WAS A LINE FEED
        001270  001366                                  BNE     1$              ;BRANCH IF NOT LINE FEED

        001272  016746  177504                          MOV     $FILLS,-(SP)    ;GET # OF FILLERS REQUIRED AFTER
                                                                                ;LINE FEED AND FILLER CHARACTER
        001276  105366  000001                  4$:     DECB    1(SP)           ;DECREMENT FILLERS COUNT
        001302  002770                                  BLT     3$              ;BRANCH IF NO MORE FILLERS NEEDED
        001304  004767  000002                          JSR     PC,5$           ;TYPE FILLER CHARACTER
        001310  000772                                  BR      4$

        001312  105737  177564                  5$:     TSTB    @#TPS           ;WAIT FOR OUTPUT DEVICE
        001316  100375                                  BPL     .-4             ;TO BECOME READY
        001320  116637  000002  177566                  MOVB    2(SR),@#TPB     ;TYPE CHARACTER
        001326  000207                                  RTS     PC

                000000                                  NULL=0
                                                ;ROUTINE TO PLACE ASCII VALUE OF AN ADDRESS IN TO ADDRESS MESSAGE
        001330                                  $FORM0:
        001330  004767  176660                          JSR     PC,$SAVR        ;GO SAVE REGISTERS ON THE STACK
        001334  012704  001662                          MOV     #DIGITS,R4      ;ADDRESS WHERE ASCII VALUES ARE STORED
        001340  005003                                  CLR     R3              ;WORKING & INDEX REGISTER
        001342  010201                                  MOV     R2,R1           ;SAVE
        001344  006302                          1$:     ASL     R2              ;FIRST DIGIT TO R3
        001346  006103                                  ROL     R3
        001350  012700  000006                          MOV     #6,R0           ;DIGIT COUNT
```

```
        001354  000404                                  BR      3$              ;PRINT FIRST DIGIT
        001356  006302                          2$:     ASL     R2
        001360  006103                                  ROL     R3
        001362  005301                                  DEC     R1
        001364  001374                                  BNE     2$
        001366  012701  000003                  3$:     MOV     #3,R1           ;DIGIT SHIFT COUNT
        001372  116324  001652                          MOVB    DIGTAB(3),(4)+  ;LOAD DIGIT INTO MESSAGE
        001376  005003                                  CLR     R3              ;CLEAR INDEX
        001400  005300                                  DEC     R0              ;DEC DIGIT COUNT
        001402  001365                                  BNE     2$
        001404  004767  176630                          JSR     PC,$RESTR       ;RESTORE REGISTERS FROM STACK
        001410  000207                                  RTS     PC              ;RETURN

                                                ;ERROR SERVICE CALLED BY TRAP (HLT) INSTRUCTION
        001412  005737  177570                  ERROR:  TST     @#SWR           ;HALT ON ERROR?
        001416  100002                                  BPL     .+6
        001420  000000                                  HALT                    ;ERROR PC IS THE TOP WORD
        001422  000002                                  RTI                     ;ON THE STACK
        001424  032737  020000  177570                  BIT     #20000,@#SWR    ;PRINT OUT DESIRED?
        001432  001073                                  BNE     1$              ;BRANCH IF NO PRINTOUT
        001434  011627                                  MOV     (SP),(PC)+      ;SAVE PC
        001436  000000                          11$:    .WORD   0               ;CONTAINS SAVED PC
        001440  016627  000002                          MOV     2(SP),(PC)+     ;GET STATUS ON TRAP
        001444  000000                          12$:    .WORD   0               ;CONTAINS STATUS (PSW) AT TIME OF TRAP
        001446  004767  176542                          JSR     PC,$SAVR        ;GO SAVE REGISTERS ON THE STACK
        001452  013702  001000                          MOV     @#ICNT,R2       ;GET PASS COUNT
        001456  004767  177646                          JSR     PC,$FORM0       ;GO TO FORMAT ROUTINE
        001462  016767  000176  000212                  MOV     DIGITS+2,PASSES ;LOAD ASCII VALUES
        001470  016767  000172  000206                  MOV     DIGITS+4,PASSES+2
        001476  004767  177530                          JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        001502  001672                                  PASCNT
        001504  016702  177726                          MOV     11$,R2          ;GET PC OF ERROR CALL
        001510  005742                                  TST     -(R2)           ;DECREMENT PC TO HLT
        001512  004767  177612                          JSR     PC,$FORM0       ;GO TO FORMAT ROUTINE
        001516  004767  177510                          JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        001522  001707                                  ERRPC
        001524  004767  177502                          JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        001530  001662                                  DIGITS
        001532  004767  177474                          JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        001536  001714                                  STATUS
        001540  016702  177700                          MOV     12$,R2          ;GET STAUS AT TIME OF ERROR
        001544  004767  177560                          JSR     PC,$FORM0       ;GO TO FORMAT ROUTINE
        001550  004767  177456                          JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        001554  001662                                  DIGITS
        001556  016702  177654                          MOV     11$,R2          ;GET PC OF ERROR
        001562  005742                                  TST     -(R2)
        001564  005737  001004                          TST     @#FACTOR
        001570  001412                                  BEQ     10$
        001572  163702  001004                          SUB     @#FACTOR,R2     ;FORM PC OF ORIGINAL CODE
        001576  004767  177526                          JSR     PC,$FORM0       ;GO TO FORMAT ROUTINE
        001602  004767  177424                          JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        001606  001721                                  ERRPC0
        001610  004767  177416                          JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        001614  001662                                  DIGITS
        001616                                 10$:
        001616  004767  176416                         JSR     PC,$RESTR       ;RESTORE REGISTERS FROM STACK
        001622  032737  002000  177570  1$:            BIT     #2000,@#SWR     ;RING BELL ON ERROR
        001630  001403                                 BEQ     2$
        001632  004767  177374                         JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        001636  001747                                 BELL
        001640  005737  177570          2$:            TST     @#SWR           ;HALT AFTER PRINT OUT
        001644  100001                                 BPL     .+4
        001646  000000                                 HALT
        001650  000002                                 RTI

                                        ;DIGIT TABLE
        001652  030460                  DIGTAB: "01
        001654  031462                          "23
        001656  032464                          "45
        001660  033466                          "67
        001662  030060  030060  030060  DIGITS: .ASCIZ  '000000 '
        001670  000040
        001672  005015                  PASCNT: .ASCII  <15><12>
        001674  044440  047103  036524          .ASCII  ' ICNT='
        001702  030060  030060     000  PASSES: .ASCIZ  '0000'
        001707     040  041520  000075  ERRPC:  .ASCIZ  ' PC='
        001714  051520  036527     000  STATUS: .ASCIZ  'PSW='
        001721     120  020103  042522  ERRPC0: .ASCIZ  'PC RELOCATED FROM '
        001726  047514  040503  042524
        001734  020104  051106  046517
        001742  000040
        001744  005015     000          $CRLF:  .ASCIZ  <15><12>
        001747     007     000          BELL:   .ASCIZ  <7>
                001752                          .EVEN

                                        ;ROUTINE TO GET TYPED OCTAL ADDRESS AND CONVERT TO OCTAL, CALL:
                                        ;       JSR     R5,RECD
                                        ;       .WORD   0               ;CONVERTED DATA IS PLACED HERE
        001752  010046                  RECD:   MOV     R0,-(SP)        ;SAVE R0 ON THE STACK
        001754  005015                          CLR     (R5)            ;CLEAR OLD DATA
        001756  105737  177560          1$:     TSTB    @#TKS           ;WAIT FOR USER TO TYPE CHARACTER
        001762  100375                          BPL     1$
        001764  113700  177562                  MOVB    @#TKB,R0        ;GET CHARACTER
        001770  042700  000200                  BIC     #200,R0         ;STRIP MSB
        001774  122700  000177                  CMPB    #177,R0         ;CHECK IF RUBOUT
        002000  001010                          BNE     2$              ;BRANCH IF NOT RUBOUT
        002002  112737  000134  177566          MOVB    #'\,@#TRB       ;TYPE \
        002010  000241                          CLC                     ;CLEAR CARRY
        002012  006015                          ROR     (R5)            ;SHIFT LAST TYPED CHARACTER
        002014  006215                          ASR     (R5)            ;OUT OF DATA WORD
        002016  006215                          ASR     (R5)
        002020  000756                          BR      1$              ;GO WAIT FOR NEXT CHARACTER

        002022  110037  177566          2$:     MOVB    R0,@#TPB        ;ECHO CHARACTER TYPED
        002026  122700  000015                  CMPB    #15,R0          ;CHECK IF CARRIAGE RETURN
        002032  001005                          BNE     3$              ;BRANCH IF NOT CARRIAGE RETURN
        002034  004767  177172                  JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        002040  001744                          $CRLF
```

```
        002042  005725                          TST     (R5)+           ;STEP RETURN ADDRESS
        002044  000205                          RTS     R5              ;RETURN

        002046  042700  177770          3$:     BIC     #177770,R0      ;STRIP NON-ESSENTIAL BITS
        002052  006315                          ASL     (R5)            ;SHIFT LAST CHARACTER 3 PLACES
        002054  006315                          ASL     (R5)            ;LEFT
        002056  006315                          ASL     (R5)
        002060  050015                          BIS     R0,(R5)         ;AND INSERT NEW CHARACTER
        002062  000735                          BR      1$              ;WAIT FOR NEXT CHARACTER

        002064  000002                          RTI                             ;RETURN
```

```
                                        .TITLE  DZQKC-C BASIC 11 FAMILY INSTRUCTION EXER.

        002066  005037  177776          START:  CLR     @#PSW           ;KERNEL MODE
        002072  005000                          CLR     R0              ;CLEAR R0-R5
        002074  005001                          CLR     R1
        002076  005002                          CLR     R2
        002100  005003                          CLR     R3
        002102  005004                          CLR     R4
        002104  005005                          CLR     R5
        002106  012706  000600                  MOV     #KPTR,SR        ;SET KERNEL STACK PTR

                                        ;ROUTINE TO DETERMINE LAST MEMORY ADDRESS
        002112  012737  002132  000004          MOV     #1$,@#ERRVEC
        002120  005037  000006                  CLR     @#ERRVEC+2
        002124  005000                          CLR     R0
        002126  005720                          TST     (R0)+           ;WILL TIME OUTWHEN END OF MEMORY
        002130  000776                          BR      .-2
        002132  162700  000002          1$:     SUB     #2,R0
        002136  010027                          MOV     R0,(PC)+        ;SET VALUE INTO LSTMEM
        002140  000000                  LSTMEM: .WORD   0               ;CONTAINS VALUE OF LAST MEMORY ADDRESS
        002142  162737  004000  002140          SUB     #4000,@#LSTMEM  ;SET PROTECTION FOR LOADERS
        002150  012737  020000  001012          MOV     #20000,@#FRSTMEM;SET LOWER BOUNDARY AT 20000
        002156  000422                          BR      START3          ;GO TO START 3
        002160                          START1:
        002160  004767  177046                  JSR     PC,,PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        002164  016576                          MSG1
        002166  004567  177560                  JSR     R5,RECD         ;GET LOWER LIMIT
        002172  000000                  1$:     .WORD   0               ;CONTAINS TYPED LOWER LIMIT
        002174  016737  177772  001012          MOV     1$,@#FRSTMEM    ;SET IN LOWER LIMIT
        002202  004767  177024                  JSR     PC,,PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        002206  016613                          MSG2
        002210  004567  177536                  JSR     R5,RECD         ;GET UPPER LIMIT
        002214  000000                  2$:     .WORD   0               ;CONTAINS UPPER LIMIT
        002216  016737  177772  002140          MOV     2$,@#LSTMEM

        002224  005037  001000          START3: CLR     @#ICNT          ;CLEAR PASS COUNT
        002230  012737  000006  000004  START2: MOV     #ERRVEC+2,@#ERRVEC      ;SET ERROR TRAP TO HALT AT 6
        002236  012706  000500                  MOV     #STKPTR,SP      ;SET STACK PTR
        002242  013737  001000  177570          MOV     @#ICNT,@#DISPLAY;DISPLAY PASS COUNT
        002250  012737  001016  000030          MOV     #SCOPEA,@#EMTVEC;SET EMT(SCOPE) TRAP VECTOR
        002256  012737  001412  000034          MOV     #ERROR,@#TRAPVEC        ;SET TRAP (HLT) VECTOR
        002264  012737  000200  000036          MOV     #200,@#TRAPVEC+2        ;PRIORITY LEVEL 4 ON TRAP

                                        ;000000000000 FIRST ADDRESS TO BE RELOCATED 00000000
        002272  010700                  REL0:   MOV     PC,R0           ;GET PC
        002274  005740                          TST     -(R0)           ;R0 CONTAINS THE ADDRESS OF REL0
        002276  010037  001010                  MOV     R0,@#FRSTAD     ;SAVE
        002302  010700                          MOV     PC,R0           ;GET CURRENT PC
        002304  162700  002304                  SUB     #.,R0           ;SUBTRACT RELOCATION FACTOR
        002310  010037  001004                  MOV     R0,@#FACTOR     ;SAVE RELOCATION FACTOR
        002314  010701                          MOV     PC,R1           ;SET NEW SCOPE PTR
                                        ;CHECK BRANCH INSTRUCTIONS
        002316  000257                          CCC                     ;CC'S=0000
        002320  103407                          BCS     CC0             ;SAME AS BLO
```

```
        002322  102406                          BVS     CC0
        002324  001405                          BEQ     CC0
        002326  100404                          BMI     CC0
        002330  002403                          BLT     CC0
        002332  003402                          BLE     CC0
        002334  101401                          BLOS    CC0
        002336  101001                          BHI     .+4
        002340  104400                  CC0:    HLT                     ;ONE OF THE ABOVE BRANCHES FAILED

                                        ;CONTINUE
        002342  000270                          SEN                     ;CC'S=1000
        002344  100003                          BPL     CC1
        002346  002002                          BGE     CC1
        002350  003001                          BGT     CC1
        002352  002401                          BLT     .+4
        002354  104400                  CC1:    HLT                     ;ONE OF THE ABOVE BRANCHES FAILED

                                        ;CONTINUE
        002356  000262                          SEV                     ;CC'S=1010
        002360  102003                          BVC     CC2
        002362  002402                          BLT     CC2
        002364  003401                          BLE     CC2
        002366  002001                          BGE     .+4
        002370  104400                  CC2:    HLT                     ;ERROR! ONE OF THE ABOVE BRANCHES FAILED

                                        ;CONTINUE
        002372  000261                          SEC                     ;CC'S=1011
        002374  103002                          BCC     CC3
        002376  101001                          BHI     CC3
        002400  003001                          BGT     .+4
        002402  104400                  CC3:    HLT                     ;ERROR! ONE OF THE ABOVE BRANCHES FAILED

                                        ;CONTINUE
        002404  000264                          SEZ                     ;CC'S=1111
        002406  001003                          BNE     CC4
        002410  003002                          BGT     CC4
        002412  101001                          BHI     CC4
        002414  003401                          BLE     .+4
        002416  104400                  CC4:    HLT                     ;ERROR! ONE OF THE ABOVE BRANCHES FAILED
        002420  104000                          SCOPE

                                        ;TEST UNARY CONDITION CODES
                                        ;CLR    R0
        002422  000277                          SCC
        002424  000244                          CLZ
        002426  005000                          CLR     R0              ;R0=0,CC'S=0100
        002430  103404                          BCS     CLR0
        002432  102403                          BVS     CLR0
        002434  001002                          BNE     CLR0
        002436  100401                          BMI     CLR0
        002440  003401                          BLE     .+4
        002442  104400                  CLR0:   HLT                     ;ERROR! INCORRECT CC'S AFTER CLR

        002444  000277                          SCC
        ØØ2446  ØØØ244                          CLE
        ØØ245Ø  ØØ57ØØ                          TST     RØ              ;RØ=Ø,CC'S=Ø1ØØ
        ØØ2452  1Ø34Ø4                          BCS     TSTØ
        ØØ2454  1Ø24Ø3                          BVS     TSTØ
        ØØ2456  ØØ1ØØ2                          BNE     TSTØ
        ØØ246Ø  1ØØ4Ø1                          BMI     TSTØ
        ØØ2462  1Ø14Ø1                          BLOS    .+4
        ØØ2464  1Ø44ØØ                  TSTØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER TST

        ØØ2466  ØØØ257                          CCC
        ØØ247Ø  ØØØ266                          +SEZ!SEV
        ØØ2472  ØØ51ØØ                          COM     RØ              ;RØ=-1,CC'S=1ØØ1
        ØØ2474  1Ø3ØØ4                          BCC     COMØ
        ØØ2476  1Ø24Ø3                          BVS     COMØ
        ØØ25ØØ  ØØ14Ø2                          BEQ     COMØ
        ØØ25Ø2  1ØØØØ1                          BPL     COMØ
        ØØ25Ø4  ØØ24Ø1                          BLT     .+4
        ØØ25Ø6  1Ø44ØØ                  COMØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER COM

        ØØ251Ø  ØØØ261                          SEC
        ØØ2512  ØØ55ØØ                          ADC     RØ              ;RØ=ØØØØØØ,CC'S=Ø1Ø1
        ØØ2514  1Ø3ØØ3                          BCC     ADCØ
        ØØ2516  1Ø24Ø2                          BVS     ADCØ
        ØØ252Ø  ØØ1ØØ1                          BNE     ADCØ
        ØØ2522  ØØ2ØØ1                          BGE     .+4
        ØØ2524  1Ø44ØØ                  ADCØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER ADC

        ØØ2526  ØØØ261                          SEC
        ØØ253Ø  ØØ6ØØØ                          ROR     RØ              ;RØ=1ØØØØØ,CC'S=1Ø1Ø
        ØØ2532  1Ø34Ø4                          BCS     RORØ
        ØØ2534  1Ø2ØØ3                          BVC     RORØ
        ØØ2536  ØØ14Ø2                          BEQ     RORØ
        ØØ254Ø  1ØØØØ1                          BPL     RORØ
        ØØ2542  ØØ3ØØ1                          BGT     .+4
        ØØ2544  1Ø44ØØ                  RORØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER ROR
        ØØ2546  ØØØ277                          SCC
        ØØ255Ø  ØØØ242                          CLV
        ØØ2552  ØØ53ØØ                          DEC     RØ                      ;RØ=Ø77777,CC'S=ØØ11
        ØØ2554  1Ø3ØØ4                          BCC     DECØ
        ØØ2556  1Ø2ØØ3                          BVC     DECØ
        ØØ256Ø  ØØ14Ø2                          BEQ     DECØ
        ØØ2562  1ØØ4Ø1                          BMI     DECØ
        ØØ2564  ØØ34Ø1                          BLE     .+4
        ØØ2566  1Ø44ØØ                  DECØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER DEC

        ØØ257Ø  ØØØ257                          CCC
        ØØ2572  ØØ52ØØ                          INC     RØ              ;RØ=1ØØØØØ,CC'S=1Ø1Ø
        ØØ2574  1Ø34Ø4                          BCS     INCØ
        ØØ2576  1Ø2ØØ3                          BVC     INCØ
        ØØ26ØØ  ØØ14Ø2                          BEQ     INCØ
        ØØ26Ø2  1ØØØØ1                          BPL     INCØ
        ØØ26Ø4  ØØ3ØØ1                          BGT     .+4
        ØØ26Ø6  1Ø44ØØ                  INCØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER INC
```

```
        ØØ261Ø  ØØØ277                          SCC
        ØØ2612  ØØØ242                          CLV
        ØØ2614  ØØ54ØØ                          NEG     RØ              ;RØ=1ØØØØØ,CC'S=1Ø11
        ØØ2616  1Ø3ØØ3                          BCC     NEGØ
        ØØ262Ø  1Ø2ØØ2                          BVC     NEGØ
        ØØ2622  ØØ14Ø1                          BEQ     NEGØ
        ØØ2624  ØØ2ØØ1                          BGE     .+4
        ØØ2626  1Ø44ØØ                  NEGØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER NEG

        ØØ263Ø  ØØØ261                          SEC
        ØØ2632  ØØ63ØØ                          ASL     RØ              ;RØ=ØØØØØØ,CC'S=Ø111
        ØØ2634  1Ø3ØØ4                          BCC     ASLØ
        ØØ2636  1Ø2ØØ3                          BVC     ASLØ
        ØØ264Ø  ØØ1ØØ2                          BNE     ASLØ
        ØØ2642  1ØØ4Ø1                          BMI     ASLØ
        ØØ2644  1Ø14Ø1                          BLOS    .+4
        ØØ2646  1Ø44ØØ                  ASLØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER ASL

        ØØ265Ø  ØØ61ØØ                          ROL     RØ              ;RØ=ØØØØØ1,CC'S=ØØØØ
        ØØ2652  1Ø34Ø2                          BCS     ROLØ
        ØØ2654  ØØ34Ø1                          BLE     ROLØ
        ØØ2656  ØØ2ØØ1                          BGE     .+4
        ØØ266Ø  1Ø44ØØ                  ROLØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER ROL

        ØØ2662  ØØ62ØØ                          ASR     RØ              ;RØ=ØØØØØØ,CC'S=Ø111
        ØØ2664  1Ø3ØØ3                          BCC     ASRØ
        ØØ2666  1Ø2ØØ2                          BVC     ASRØ
        ØØ267Ø  ØØ1ØØ1                          BNE     ASRØ
        ØØ2672  ØØ24Ø1                          BLT     .+4
        ØØ2674  1Ø44ØØ                  ASRØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER ASR

        ØØ2676  ØØØ277                          SCC
        ØØ27ØØ  ØØ56ØØ                          SBC     RØ              ;RØ=-1,CC'S=1ØØ1
        ØØ27Ø2  1Ø3ØØ2                          BCC     SBCØ
        ØØ27Ø4  1Ø24Ø1                          BVS     SBCØ
        ØØ27Ø6  ØØ34Ø1                          BLE     .+4
        ØØ271Ø  1Ø44ØØ                  SBCØ:   HLT                     ;ERROR! INCORRECT CC'S AFTER SBC

        ØØ2712  ØØ54ØØ                          NEG     RØ              ;RØ=ØØØØØ1,CC'S=ØØØØ1
        ØØ2714  ØØØ3ØØ                          SWAB    RØ              ;RØ=ØØØ4ØØ ,CC'S=Ø1ØØ
        ØØ2716  1Ø34Ø3                          BCS     SWABØ
        ØØ272Ø  1Ø24Ø2                          BVS     SWABØ
        ØØ2722  ØØ1ØØ1                          BNE     SWABØ
        ØØ2724  ØØ2ØØ1                          BGE     .+4
        ØØ2726  1Ø44ØØ                  SWABØ:  HLT                     ;ERROR! INCORRECT CC'S AFTER SWAB
        ØØ273Ø  1Ø4ØØØ                          SCOPE

                                        ;CHECK REGISTER SELECTION
        ØØ2732  ØØ5ØØØ                          CLR     RØ
        ØØ2734  ØØØ277                          SCC
        ØØ2736  ØØ61ØØ                          ROL     RØ              ;RØ=1
        ØØ274Ø  Ø1ØØØ2                          MOV     RØ,R2
        ØØ2742  ØØ63Ø2                          ASL     R2              ;R2=2
        ØØ2744  Ø1Ø2Ø3                          MOV     R2,R3
        002746  006303                          ASL     R3              ;R3=4
        002750  010304                          MOV     R3,R4
        002752  006304                          ASL     R4              ;R4=10
        002754  010405                          MOV     R4,R5
        002756  006305                          ASL     R5              ;R5=20
        002760  010546                          MOV     R5,-(SP)        ;SET BITS SET IN REGISTERS
        002762  050416                          BIS     R4,(SP)         ;INTO STACK ADDRESS
        002764  050316                          BIS     R3,(SP)
        002766  050216                          BIS     R2,(SP)
        002770  050016                          BIS     R0,(SP)
        002772  022726  000037                  CMP     #37,(SP)+
        002776  001401                          BEQ     .+4             ;WERE SET
        003000  104400                          HLT                     ;MISSING BIT(S) REPRESENT
                                                                        ;INCORRECT REGISTER SELECTION

                                        ;CHECK THAT ALL BITS CAN BE SET & CLEARED IN ALL REGISTERS
        003002  000257                          CCC
        003004  112700  000377                  MOVB    #377,R0         ;SET ALL BITS (MOVB EXTENDS SIGN)
        003010  006100                  1$:     ROL     R0                      ;ROTATE A 0 THROUGH ALL BIT
        003012  103776                          BCS     1$                      ;POSITIONS
        003014  005200                          INC     R0                      ;FINAL RESULT IS -1
        003016  001401                          BEQ     .+4
        003020  104400                          HLT                             ;ERROR!

        003022  012700  000020                  MOV     #16.,R0         ;SET SHIFT COUNT
        003026  005002                          CLR     R2
        003030  000261                  2$:     SEC
        003032  006002                          ROR     R2              ;ROTATE 1 THROUGH ALL BIT POSITS
        003034  005300                          DEC     R0              ;DECREMENT SHIFT COUNT
        003036  001374                          BNE     2$
        003040  005102                          COM     R2              ;R2 SHOULD CONTAIN -1
        003042  001401                          BEQ     .+4
        003044  104400                          HLT                     ;ERROR! CHECK R2 SHOULD = 0

        003046  012703  100000                  MOV     #100000,R3
        003052  006203                  3$:     ASR     R3              ;EXTEND 1 BIT THROUGH ALL POSITIONS
        003054  103376                          BCC     3$
        003056  005203                          INC     R3
        003060  001401                          BEQ     .+4
        003062  104400                          HLT                             ;ERROR!

        003064  112704  177401                  MOVB    #177401,R4      ;R4=1
        003070  060404                  4$:     ADD     R4,R4           ;HAS THE AFFECT OF SHIFTING A BIT
        003072  103376                          BCC     4$              ;THROUGH ALL POSITIONS
        003074  005704                          TST     R4              ;RESULT SHOULD BE 0
        003076  001401                          BEQ     .+4
        003100  104400                          HLT

        003102  012705  000001                  MOV     #1,R5
        003106  006305                  5$:     ASL     R5
        003110  102376                          BVC     5$
        003112  006305                          ASL     R5
        003114  103002                          BCC     6$
        003116  005705                          TST     R5
```

```
        003120  001401                          BEQ     .+4
        003122  104400                  6$:     HLT

                                        ;CHECK REGISTER VOLITILITY
        003124  005002                          CLR     R2
        003126  005102                          COM     R2              ;R2=-1
        003130  010203                          MOV     R2,R3
        003132  000257                          CCC
        003134  006002                          ROR     R2              ;R2=LOOP COUNT
        003136  006202                          ASR     R2
        003140  010304                  7$:     MOV     R3,R4
        003142  005302                          DEC     R2              ;DECREMENT LOOP COUNT
        003144  001375                          BNE     7$
        003146  005203                          INC     R3              ;CHECK R3
        003150  001002                          BNE     8$
        003152  005204                          INC     R4              ;CHECK R4
        003154  001401                          BEQ     .+4
        003156  104400                  8$:     HLT

                                        ;CHECK TRANSFER OF REGISTER DATA BETWEEN THE GS AND GD REGISTERS (11/45)
        003160  032737  000020  177776  GSTST:  BIT     #20,@#PSW       ;CHECK IF 'T' BIT IS SET
        003166  001052                          BNE     7$              ;SKIP TEST IF 'T' BIT SET
        003170  010146                          MOV     R1,-(SP)        ;SAVE SCOPE PTR
        003172  010627                          MOV     SP,(PC)+        ;SAVE STACK PTR
        003174  000000                  1$:     .WORD   0               ;CONTAINS SAVED STACK PTR
        003176  010727                          MOV     PC,(PC)+        ;LOAD DATA, THE CURRENT PC IS USED AS
        003200  000000                  2$:     .WORD   0               ;DATA, IF THIS TEST FAILS 2$ CON-
                                                                        ;TAINS THE DATA BEING USED,
        003202  005267  177772                  INC     2$              ;MAKE ODD TO CHECK BIT 0
        003206  016700  177766          3$:     MOV     2$,R0           ;LOAD GD REGISTER 0
        003212  010001                          MOV     R0,R1           ;TRANSFER GS REG 0 TO GD REG 1
        003214  010102                          MOV     R1,R2           ;AND GS REG 1 TO GD REG 2
        003216  010203                          MOV     R2,R3           ;ETC...
        003220  010304                          MOV     R3,R4
        003222  010405                          MOV     R4,R5
        003224  152737  000340  177776          BISB    #340,@#RSW      ;SET PRIORITY LEVEL 7
        003232  010506                          MOV     R5,SP           ;TRANSFER GS REG 5 TO GD STK PTR
        003234  010627                          MOV     SP,(PC)+        ;TRANSFER GS STK PTR TO MEMORY
        003236  000000                  4$:     .WORD   0               ;CONTAINS GS STACK PTR
        003240  016706  177730                  MOV     1$,SP           ;RESTORE STK PTR NEEDED FOR HLT/SCOPE
        003244  142737  000340  177776          BICB    #340,@#RSW      ;SET PRIORITY LEVEL 0
        003252  026700  177760                  CMP     4$,R0           ;COMPARE GS/GD STKPTR WITH GS REG 0
        003256  001004                          BNE     5$              ;BRANCH IF THEY WERE NOT =
        003260  006367  177714                  ASL     2$              ;SHIFT TEST DATA UNTIL = 000000
        003264  001350                          BNE     3$
        003266  000411                          BR      6$
        003270  010046                  5$:     MOV     R0,-(SP)        ;GET GS REG 0
        003272  010146                          MOV     R1,-(SP)        ;ETC...
        003274  010246                          MOV     R2,-(SP)
        003276  010346                          MOV     R3,-(SP)
        003300  010446                          MOV     R4,-(SP)
        003302  010546                          MOV     R5,-(SP)
        003304  104400                          HLT                     ;ERROR! DATA IN GS STK PTR NOT = GS REG 0
                                                                        ;GS REG 0-GS REG 5 ARE ON THE STACK
        ØØ33Ø6  Ø167Ø6  177662                  MOV     1$,SP           ;RESTORE STACK PTR
        ØØ3312  Ø126Ø1                  6$:     MOV     (SP)+,R1        ;RESTORE SCOPE PTR
        ØØ3314  1Ø4ØØØ                  7$:     SCOPE

                                        ;TEST UNARY WORD INSTRUCTIONS USING ADDRESS MODE 1
        ØØ3316  ØØØ4Ø1                          BR      .+4
        ØØ332Ø  ØØØØØØ                          .WORD   Ø               ;RESERVE ADDRESS FOR TESTS
        ØØ3322  Ø1Ø7Ø2                          MOV     PC,R2
        ØØ3324  1627Ø2  ØØØØØ4                  SUB     #4,R2           ;R2 POINTS TO RESERVED WORD
        ØØ333Ø  ØØ5Ø12                          CLR     (R2)            ;PRESET (R2)

        ØØ3332  ØØØ261                          SEC
        ØØ3334  ØØ6Ø12                          ROR     (R2)            ;(R2)=1ØØØØØ,CC=1Ø1Ø
        ØØ3336  1Ø14Ø2                          BLOS    ROR1
        ØØ334Ø  1ØØØØ1                          BPL     ROR1
        ØØ3342  ØØ2ØØ1                          BGE     .+4
        ØØ3344  1Ø44ØØ                  ROR1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        ØØ3346  ØØØ257                          CCC
        ØØ335Ø  ØØØ261                          SEC
        ØØ3352  ØØ5312                          DEC     (R2)            ;(R2)=Ø77777,CC=ØØ11
        ØØ3354  1Ø3ØØ1                          BCC     DEC1
        ØØ3356  ØØ34Ø1                          BLE     .+4
        ØØ336Ø  1Ø44ØØ                  DEC1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        ØØ3362  ØØØ257                          CCC
        ØØ3364  ØØØ261                          SEC
        ØØ3366  ØØ5512                          ADC     (R2)            ;(R2)=1ØØØØØ,CC=1Ø1Ø
        ØØ337Ø  1Ø34Ø3                          BCS     ADC1
        ØØ3372  1Ø2ØØ2                          BVC     ADC1
        ØØ3374  1ØØØØ1                          BPL     ADC1
        ØØ3376  ØØ1ØØ1                          BNE     .+4
        ØØ34ØØ  1Ø44ØØ                  ADC1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        ØØ34Ø2  ØØ6112                          ROL     (R2)            ;(R2)=ØØØØØØ,CC=Ø111
        ØØ34Ø4  1Ø3ØØ3                          BCC     ROL1
        ØØ34Ø6  1Ø2ØØ2                          BVC     ROL1
        ØØ341Ø  ØØ1ØØ1                          BNE     ROL1
        ØØ3412  1ØØØØ1                          BPL     .+4
        ØØ3414  1Ø44ØØ                  ROL1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        ØØ3416  ØØ6112                          ROL     (R2)            ;(R2)=ØØØØØ1,CC=ØØØØ
        ØØ342Ø  1Ø14Ø2                          BLOS    ROL1A           ;BRANCH IF C OR Z IS SET
        ØØ3422  1Ø24Ø1                          BVS     ROL1A
        ØØ3424  1ØØØØ1                          BPL     .+4
        ØØ3426  1Ø44ØØ                  ROL1A:  HLT

        ØØ343Ø  ØØ6212                          ASR     (R2)            ;(R2)=ØØØØØØ,CC=Ø111
        ØØ3432  1Ø3ØØ3                          BCC     ASR1
        ØØ3434  1Ø2ØØ2                          BVC     ASR1
        ØØ3436  ØØ1ØØ1                          BNE     ASR1
        ØØ344Ø  1ØØØØ1                          BPL     .+4
        ØØ3442  1Ø44ØØ                  ASR1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE
```

```
        ØØ3444  ØØ6Ø12                          ROR     (R2)            ;(R2)=1ØØØØØ,CC=1Ø1Ø
        ØØ3446  1Ø34Ø3                          BCS     ROR1A
        ØØ345Ø  1Ø2ØØ2                          BVC     ROR1A
        ØØ3452  ØØ14Ø1                          BEQ     ROR1A
        ØØ3454  1ØØ4Ø1                          BMI     .+4
        ØØ3456  1Ø44ØØ                  ROR1A:  HLT

        ØØ346Ø  ØØØ261                          SEC
        ØØ3462  ØØ5212                          INC     (R2)            ;(R2)=1ØØØØ1,CC=1ØØ1
        ØØ3464  1Ø3ØØ3                          BCC     INC1
        ØØ3466  1Ø24Ø2                          BVS     INC1
        ØØ347Ø  ØØ14Ø1                          BEQ     INC1
        ØØ3472  1ØØ4Ø1                          BMI     .+4
        ØØ3474  1Ø44ØØ                  INC1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        ØØ3476  ØØ5612                          SBC     (R2)            ;(R2)=1ØØØØØ,CC=1ØØØ
        ØØ35ØØ  1Ø34Ø3                          BCS     SBC1
        ØØ35Ø2  1Ø24Ø2                          BVS     SBC1
        ØØ35Ø4  ØØ14Ø1                          BEQ     SBC1
        ØØ35Ø6  1ØØ4Ø1                          BMI     .+4
        ØØ351Ø  1Ø44ØØ                  SBC1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        ØØ3512  ØØØ261                          SEC
        ØØ3514  ØØ5612                          SBC     (R2)            ;(R2)=Ø77777,CC=ØØ1Ø
        ØØ3516  1Ø34Ø3                          BCS     SBC1A
        ØØ352Ø  1Ø2ØØ2                          BVC     SBC1A
        ØØ3522  ØØ14Ø1                          BEQ     SBC1A
        ØØ3524  1ØØØØ1                          BPL     .+4
        ØØ3526  1Ø44ØØ                  SBC1A:  HLT                     ;ERROR! INCORRECT CC'S AS SHOEN ABOVE

        ØØ353Ø  ØØØ261                          SEC
        ØØ3532  ØØ5512                          ADC     (R2)            ;(R2)=1ØØØØØ,CC=1Ø1Ø
        ØØ3534  1ØØ4Ø1                          BMI     .+4
        ØØ3536  1Ø44ØØ                          HLT

        ØØ354Ø  ØØØ261                          SEC
        ØØ3542  ØØ6312                          ASL     (R2)            ;(R2)=ØØØØØØ,CC=Ø111
        ØØ3544  1Ø3ØØ3                          BCC     ASL1
        ØØ3546  1Ø2ØØ2                          BVC     ASL1
        ØØ355Ø  ØØ1ØØ1                          BNE     ASL1
        ØØ3552  1ØØØØ1                          BPL     .+4
        ØØ3554  1Ø44ØØ                  ASL1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        ØØ3556  ØØ5112                          COM     (R2)            ;(R2)=177777,CC=1ØØ1
        ØØ356Ø  1Ø3ØØ2                          BCC     COM1
        ØØ3562  1Ø24Ø1                          BVS     COM1
        ØØ3564  1ØØ4Ø1                          BMI     .+4
        ØØ3566  1Ø44ØØ                  COM1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        ØØ357Ø  ØØØ25Ø                          CLN
        ØØ3572  ØØ5712                          TST     (R2)            ;(R2)=177777,CC=1ØØØ
        ØØ3574  1Ø34Ø3                          BCS     TST1
        ØØ3576  1Ø24Ø2                          BVS     TST1
        ØØ36ØØ  1ØØØØ1                          BPL     TST1
        003602  001001                          BNE     .+4
        003604  104400                  TST1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        003606  000262                          SEV
        003610  005412                          NEG     (R2)            ;(R2)=000001,CC=0000
        003612  103002                          BCC     NEG1
        003614  102401                          BVS     NEG1
        003616  001001                          BNE     .+4
        003620  104400                  NEG1:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        003622  005312                          DEC     (R2)            ;(R2)=000000,CC=0101
        003624  103001                          BCC     DEC1A
        003626  001401                          BEQ     .+4
        003630  104400                  DEC1A:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE
        003632  104000                          SCOPE

                                        ;CHECK UNARY BYTE INSTRUCTIONS USING ADDRESS MODE 1
        003634  000401                          BR      .+4             ;RESERVE A WORD
        003636  000000                          .WORD   0               ;ADDRESS RESERVED FOR TESTS
        003640  010703                          MOV     PC,R3
        003642  162703  000004                  SUB     #4,R3           ;R3 POINTS TO EVEN BYTE OF WORD
        003646  010304                          MOV     R3,R4           ;R4 POINTS TO ODD BYTE OF WORD
        003650  005204                          INC     R4
        003652  005013                          CLR     (R3)            ;PRESET DATA

        003654  000261                  1$:     SEC
        003656  105513                          ADCB    (R3)            ;ADD CARRY TO EVEN BYTE
        003660  100402                          BMI     2$              ;UNTIL EVEN BYTE BECOMES NEGATIVE
        003662  105214                          INCB    (R4)            ;INCREMENT ODD BYTE
        003664  000773                          BR      1$
        003666  102401                  2$:     BVS     .+4             ;(R3)=077600=[0774][200],CC=1010
        003670  104400                          HLT
        003672  000242                          CLV
        003674  105214                          INCB    (R4)            ;(R3)=100200=[1000][200],CC=1010
        003676  103402                          BCS     INCB1
        003700  102001                          BVC     INCB1
        003702  100401                          BMI     .+4
        003704  104400                  INCB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        003706  106114                          ROLB    (R4)            ;(R3)=000200=[0000][200],CC=0111
        003710  103002                          BCC     ROLB1
        003712  102001                          BVC     ROLB1
        003714  001401                          BEQ     .+4
        003716  104400                  ROLB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        003720  105614                          SBCB    (R4)            ;(R3)=177600=[1774][200], CC=1001
        003722  103002                          BCC     SBCB1
        003724  102401                          BVS     SBCB1
        003726  100401                          BMI     .+4
        003730  104400                  SBCB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        003732  106313                          ASLB    (R3)            ;(R3)=177400,CC=0111
        003734  103002                          BCC     ASLB1
        003736  102001                          BVC     ASLB1
```

```
        003740  001401                          BEQ     .+4
        003742  104400                  ASLB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        003744  105413                          NEGB    (R3)            ;(R3)=177400,CC=0100
        003746  103402                          BCS     NEGB1
        003750  102401                          BVS     NEGB1
        003752  001401                          BEQ     .+4
        003754  104400                  NEGB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        003756  000277                          SCC
        003760  105313                          DECB    (R3)            ;(R3)=177777,CC=1001
        003762  103002                          BCC     DECB1
        003764  102401                          BVS     DECB1
        003766  001001                          BNE     .+4
        003770  104400                  DECB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        003772  000241                          CLC
        003774  106013                          RORB    (R3)            ;(R3)=177577,CC=0011
        003776  103002                          BCC     RORB1
        004000  102001                          BVC     RORB1
        004002  100001                          BPL     .+4
        004004  104400                  RORB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004006  000241                          CLC
        004010  105114                          COMB    (R4)            ;(R3)=000177,CC=0101
        004012  103002                          BCC     COMB1
        004014  102401                          BVS     COMB1
        004016  001401                          BEQ     .+4
        004020  104400                  COMB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004022  106213                  1$:     ASRB    (R3)            ;SHIFT EVEN BYTE UNTIL V CLEARS
        004024  102002                          BVC     2$
        004026  105514                          ADCB    (R4)            ;AND ADD CARRY TO ODD BYTE
        004030  000774                          BR      1$
        004032  103401                  2$:     BCS     ASRB1
        004034  001401                          BEQ     .+4
        004036  104400                  ASRB1:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004040  106214                          ASRB    (R4)
        004042  106214                          ASRB    (R4)            ;(R3)=000400,CC=0011
        004044  103002                          BCC     ASRB1A
        004046  102001                          BVC     ASRB1A
        004050  001001                          BNE     .+4
        004052  104400                  ASRB1A: HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004054  105314                          DECB    (R4)            ;(R3)=000000,CC=0100
        004056  001401                          BEQ     .+4
        004060  104400                          HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004062  000261                          SEC
        004064  106014                          RORB    (R4)            ;(R3)=100000,CC=1010
        004066  103402                          BCS     RORB1A
        004070  102001                          BVC     RORB1A
        004072  100401                          BMI     .+4
        004074  104400                  RORB1A: HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004076  000242                          CLV
        004100  105314                          DECB    (R4)            ;(R3)=077400,CC=0100
        004102  102401                          BVS     .+4
        004104  104400                          HLT

        004106  000261                          SEC
        004110  105313                          DECB    (R3)            ;(R3)=077777,CC=1001
        004112  103002                          BCC     DECB1A
        004114  102401                          BVS     DECB1A
        004116  100401                          BMI     .+4
        004120  104400                  DECB1A: HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004122  000277                          SCC
        004124  000313                          SWAB    (R3)            ;(R3)=177577=[1774][177],CC=0000
        004126  103402                          BCS     SWAB1
        004130  102401                          BVS     SWAB1
        004132  100001                          BPL     .+4
        004134  104400                  SWAB1:  HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004136  105714                          TSTB    (R4)            ;(R3)=177577=[1774][177],CC=1000
        004140  103402                          BCS     TSTB1
        004142  102401                          BVS     TSTB1
        004144  100401                          BMI     .+4
        004146  104400                  TSTB1:  HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004150  105014                          CLRB    (R4)            ;(R3)=000177=[0000][177],CC=0100
        004152  001401                          BEQ     .+4
        004154  104400                          HLT
        004156  106313                          ASLB    (R3)            ;(R3)=000376 ,CC=1010
        004160  103402                          BCS     ASLB1A
        004162  102001                          BVC     ASLB1A
        004164  100401                          BMI     .+4
        004166  104400                  ASLB1A: HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004170  105113                          COMB    (R3)            ;(R3)=000001,CC=0001
        004172  103002                          BCC     COMB1A
        004174  102401                          BVS     COMB1A
        004176  100001                          BPL     .+4
        004200  104400                  COMB1A: HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004202  000313                          SWAB    (R3)            ;(R3)=000400, CC=0100
        004204  001401                          BEQ     .+4
        004206  104400                          HLT

        004210  105213                          INCB    (R3)
        004212  000261                          SEC
        004214  105613                          SBCB    (R3)            ;(R3)=000400,CC=0100
        004216  001401                          BEQ     .+4
        004220  104400                          HLT
        004222  022713  000400                  CMP     #400,(R3)       ;CHECK REMAINING RESULT
        004226  001401                          BEQ     .+4
        004230  104400                          HLT
```

```
        004232  104000                          SCOPE

                                        ;CHECK UNARY WORD OPS USING ADDRESS MODES 2 AND 4 (AUTO INC/DEC)
        004234  000401                          BR      .+4
        004236  000000                          .WORD   0               ;ADDRESS RESERVED FOR TESTS
        004240  010704                          MOV     PC,R4
        004242  162704  000004                  SUB     #4,R4           ;R4 AND R5 POINT TO
        004246  010405                          MOV     R4,R5           ;RESERVED WORD
        004250  005015                          CLR     (R5)            ;PRESET DATA=0

        004252  000277                          SCC
        004254  000244                          CLZ
        004256  005725                          TST     (R5)+           ;(R5)=000000,CC=0100
        004260  103402                          BCS     TST2
        004262  102401                          BVS     TST2
        004264  001401                          BEQ     .+4
        004266  104400                  TST2:   HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004270  005145                          COM     -(R5)           ;(R5)=177777,CC=1001
        004272  103001                          BCC     COM4
        004274  100401                          BMI     .+4
        004276  104400                  COM4:   HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004300  000241                          CLC
        004302  006024                          ROR     (R4)+           ;(R4)=077777,CC=0011
        004304  103002                          BCC     ROR2
        004306  102001                          BVC     ROR2
        004310  100001                          BPL     .+4
        004312  104400                  ROR2:   HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004314  000257                          CCC
        004316  005244                          INC     -(R4)           ;(R4)=100000,CC=1010
        004320  102002                          BVC     INC4
        004322  001401                          BEQ     INC4
        004324  100401                          BMI     .+4
        004326  104400                  INC4:   HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004330  000261                          SEC
        004332  000324                          SWAB    (R4)+           ;(R4)=000200,CC=1000
        004334  103401                          BCS     SWAB2
        004336  100401                          BMI     .+4
        004340  104400                  SWAB2:  HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004342  005425                          NEG     (R5)+           ;(R5)=177600,CC=1001
        004344  103001                          BCC     NEG2
        004346  100401                          BMI     .+4
        004350  104400                  NEG2:   HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        004352  005044                          CLR     -(R4)           ;(R4)=000000,CC=0100
        004354  001401                          BEQ     .+4
        004356  104400                          HLT

        004360  000261                          SEC
        004362  006045                          ROR     -(R5)           ;(R5)=100000,CC=1010
        004364  000261                          SEC
        004366  005525                          ADC     (R5)+           ;(R5)=100001,CC=1000
        004370  102401                          BVS     ADC2
        004372  100401                          BMI     .+4
        004374  104400                  ADC2:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004376  000262                          SEV
        004400  006224                          ASR     (R4)+           ;(R4)=140000,CC=1001
        004402  103002                          BCC     ASR2
        004404  102401                          BVS     ASR2
        004406  100401                          BMI     .+4
        004410  104400                  ASR2:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004412  000262                          SEV
        004414  006144                          ROL     -(R4)           ;(R4)=100001, CC=1001
        004416  103002                          BCC     ROL4
        004420  102401                          BVS     ROL4
        004422  100401                          BMI     .+4
        004424  104400                  ROL4:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004426  005645                          SBC     -(R5)           ;(R5)=100000,CC=1000
        004430  103001                          BCC     .+4
        004432  104400                          HLT                     ;ERROR! 'C' BIT FAILED TO CLEAR

        004434  005325                          DEC     (R5)+           ;(R5)=077777,CC=0010
        004436  103402                          BCS     DEC2
        004440  102001                          BVC     DEC2
        004442  100001                          BPL     .+4
        004444  104400                  DEC2:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004446  006324                          ASL     (R4)+           ;(R4)=177776,CC=1010
        004450  102401                          BVS     .+4
        004452  104400                          HLT
        004454  006344                          ASL     -(R4)           ;(R4)=177774,CC=1001
        004456  103003                          BCC     ASL4
        004460  102402                          BVS     ASL4
        004462  001401                          BEQ     ASL4
        004464  100401                          BMI     .+4
        004466  104400                  ASL4:   HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004470  022724  177774                  CMP     #177774,(R4)+
        004474  001401                          BEQ     .+4
        004476  104400                          HLT
        004500  020405                          CMP     R4,R5
        004502  001401                          BEQ     .+4
        004504  104400                          HLT
        004506  104000                          SCOPE

                                        ;CHECK UNARY BYTE OPS USING ADDRESS MODES 2 AND 4
        004510  000401                          BR      .+4             ;RESERVE A WORD
        004512  000000                          .WORD   0               ;RESERVED WORD
        004514  010705                          MOV     PC,R5
        004516  162705  000004                  SUB     #4,R5           ;R5 POINTS TO EVEN BYTE OF RESERVED WORD
        004522  010500                          MOV     R5,R0
```

```
        004524  010002                          MOV     R0,R2
        004526  005202                          INC     R2              ;R2 POINTS TO ODD BYTE OF RESERVED WORD
        004530  005010                          CLR     (R0)            ;PRESET

        004532  000277                          SCC
        004534  000241                          CLC
        004536  105125                          COMB    (R5)+           ;(R0)=000377,CC=1001
        004540  103002                          BCC     COMB2
        004542  102401                          BVS     COMB2
        004544  100401                          BMI     .+4
        004546  104400                  COMB2:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004550  105542                          ADCB    -(R2)           ;(R0)=000000,CC=0101
        004552  001401                          BEQ     .+4
        004554  104400                          HLT                     ;ERROR! INCORRECT RESULT AS SHOWN ABOVE
        004556  105525                          ADCB    (R5)+           ;(R0)=000400,CC=0000
        004560  103401                          BCS     ADCB2
        004562  001201                          BNE     .+4
        004564  104400                  ADCB2:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004566  000263                          +SEC!SEV
        004570  106045                          RORB    -(R5)           ;(R0)=100000,CC=1001
        004572  103003                          BCC     RORB4
        004574  102402                          BVS     RORB4
        004576  001401                          BEQ     RORB4
        004600  100401                          BMI     .+4
        004602  104400                  RORB4:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004604  000277                          SCC
        004606  106122                          ROLB    (R2)+           ;(R0)=100001,CC=0000
        004610  103403                          BCS     ROLB2
        004612  102402                          BVS     ROLB2
        004614  001401                          BEQ     ROLB2
        004616  100001                          BPL     .+4
        004620  104400                  ROLB2:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004622  000257                          CCC
        004624  106225                          ASRB    (R5)+           ;(R0)=140001, CC=1010
        004626  103402                          BCS     ASRB2
        004630  102001                          BVC     ASRB2
        004632  100401                          BMI     .+4
        004634  104400                  ASRB2:  HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004636  105242                          INCB    -(R2)           ;(R0)=140002,CC=0000
        004640  000277                          SCC
        004642  106222                          ASRB    (R2)+           ;(R0)=140001,CC=0000
        004644  103402                          BCS     ASRB2A
        004646  102401                          BVS     ASRB2A
        004650  100001                          BPL     .+4
        004652  104400                  ASRB2A! HLT                     ;ERROR! INCORRECT CC'S AS SHOWN ABOVE

        004654  000266                          +SEZ!SEV                ;SET Z,V
        004656  106345                          ASLB    -(R5)           ;(R0)=100001,CC=1001
        004660  103003                          BCC     ASLB4
        ØØ4662  1Ø24Ø2                          BVS     ASLB4
        ØØ4664  ØØ14Ø1                          BEQ     ASLB4
        ØØ4666  1ØØ4Ø1                          BMI     .+4
        ØØ467Ø  1Ø44ØØ                  ASLB4:  HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        ØØ4672  105322                          DECB    (R2)+           ;(RØ)=Ø774Ø1=[Ø774][ØØ1] ,CC=ØØ1Ø
        ØØ4674  1Ø3ØØ2                          BCC     DECB2
        ØØ4676  1Ø2ØØ1                          BVC     DECB2
        ØØ47ØØ  1ØØØØ1                          BPL     .+4
        ØØ47Ø2  1Ø44ØØ                  DECB2:  HLT                     ;ERROR: INCORRECT CC'S AS SHOWNABOVE

        ØØ47Ø4  105645                          SBCB    -(R5)           ;(RØ)=Ø774ØØ, CC=Ø1ØØ
        ØØ47Ø6  1Ø34Ø2                          BCS     SBCB4
        ØØ471Ø  1Ø24Ø1                          BVS     SBCB4
        ØØ4712  ØØ14Ø1                          BEQ     .+4
        ØØ4714  1Ø44ØØ                  SBCB4:  HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        ØØ4716  105442                          NEGB    -(R2)           ;(RØ)=1Ø4ØØ,CC=1ØØ1
        ØØ472Ø  1Ø3ØØ2                          BCC     NEGB4
        ØØ4722  1Ø24Ø1                          BVS     NEGB4
        ØØ4724  1ØØ4Ø1                          BMI     .+4
        ØØ4726  1Ø44ØØ                  NEGB4:  HLT                     ;ERROR: INCORRECT CC'S AS SHOWN ABOVE

        ØØ473Ø  105725                          TSTB    (R5)+           ;(RØ)=1ØØ4ØØ,CC=Ø1ØØ
        ØØ4732  1Ø34Ø1                          BCS     TSTB2
        ØØ4734  ØØ14Ø1                          BEQ     .+4
        ØØ4736  1Ø44ØØ                  TSTB2:  HLT

        ØØ474Ø  105722                          TSTB    (R2)+           ;(RØ)=1ØØ4ØØ,CC=1ØØØ
        ØØ4742  ØØ14Ø1                          BEQ     TSTB2A
        ØØ4744  1ØØ4Ø1                          BMI     .+4
        ØØ4746  1Ø44ØØ                  TSTB2A: HLT

        ØØ475Ø  ØØØ261                          SEC
        ØØ4752  ØØØ342                          SWAB    -(R2)           ;(RØ)=ØØØ2Ø1,CC=1ØØØ
        ØØ4754  1Ø34Ø1                          BCS     SWAB4
        ØØ4756  1ØØ4Ø1                          BMI     .+4
        ØØ476Ø  1Ø44ØØ                  SWAB4:  HLT

        ØØ4762  ØØØ277                          SCC
        ØØ4764  105225                          INCB    (R5)+           ;(RØ)=ØØØ6Ø1=[ØØØ4][2Ø1],CC=ØØØØ
        ØØ4766  1Ø3ØØ3                          BCC     INCB2
        ØØ477Ø  1Ø24Ø2                          BVS     INCB2
        ØØ4772  ØØ14Ø1                          BEQ     INCB2
        ØØ4774  1ØØØØ1                          BPL     .+4
        ØØ4776  1Ø44ØØ                  INCB2:  HLT

        ØØ5ØØØ  Ø22227  ØØØ6Ø1                  CMP     (R2)+,#ØØØ6Ø1   ;CHECK END RESULT
        ØØ5ØØ4  ØØ14Ø1                          BEQ     .+4
        ØØ5ØØ6  1Ø44ØØ                          HLT
        ØØ5Ø1Ø  Ø2Ø2Ø5                          CMP     R2,R5           ;CHECK REGISTERS
        ØØ5Ø12  ØØ14Ø1                          BEQ     .+4
        ØØ5Ø14  1Ø44ØØ                          HLT
        ØØ5Ø16  1Ø4ØØØ                          SCOPE
```

```
                                        ;CHECK UNARY WORD OPS USING ADDRESS MODES 3 AND 5
        ØØ5Ø2Ø  ØØØ4Ø2                          BR      .+6             ;RESERVE 2 WORDS
        ØØ5Ø22  ØØØØØØ                          .WORD   Ø               ;1 FOR THE ADDRESS
        ØØ5Ø24  ØØØØØØ                          .WORD   Ø               ;AND 1 FOR DATA
        ØØ5Ø26  Ø1Ø7Ø3                          MOV     PC,R3
        ØØ5Ø3Ø  1627Ø3  ØØØØØ4                  SUB     #4,R3
        ØØ5Ø34  ØØ5Ø13                          CLR     (R3)            ;PRESET DATA
        ØØ5Ø36  Ø1Ø3ØØ                          MOV     R3,RØ           ;RØ POINTS TO DATA WORD
        ØØ5Ø4Ø  ØØ5743                          TST     -(R3)
        ØØ5Ø42  Ø1ØØ13                          MOV     RØ,(R3)
        ØØ5Ø44  Ø1Ø3Ø4                          MOV     R3,R4

        ØØ5Ø46  ØØØ257                          CCC
        ØØ5Ø5Ø  ØØ5733                          TST     @(R3)+          ;(RØ)=ØØØØØØ,CC=Ø1ØØ
        ØØ5Ø52  ØØ14Ø1                          BEQ     .+4
        ØØ5Ø54  1Ø44ØØ                          HLT

        ØØ5Ø56  ØØØ261                          SEC
        ØØ5Ø6Ø  ØØ6Ø53                          ROR     @-(R3)          ;(RØ)=1ØØØØØ,CC=1Ø1Ø
        ØØ5Ø62  1Ø34Ø2                          BCS     ROR5
        ØØ5Ø64  1Ø2ØØ1                          BVC     ROR5
        ØØ5Ø66  1ØØ4Ø1                          BMI     .+4
        ØØ5Ø7Ø  1Ø44ØØ                  ROR5:   HLT

        ØØ5Ø72  ØØØ257                          CCC
        ØØ5Ø74  ØØ6234                          ASR     @(R4)+          ;(RØ)=14ØØØØ,CC=1Ø1Ø
        ØØ5Ø76  1Ø2ØØ1                          BVC     ASR3
        ØØ51ØØ  1ØØ4Ø1                          BMI     .+4
        ØØ51Ø2  1Ø44ØØ                  ASR3:   HLT

        ØØ51Ø4  ØØØ25Ø                          CLN
        ØØ51Ø6  ØØ6333                          ASL     @(R3)+          ;(RØ)=1ØØØØØ,CC=1ØØ1
        ØØ511Ø  1Ø3ØØ2                          BCC     ASL3
        ØØ5112  1Ø24Ø1                          BVS     ASL3
        ØØ5114  1ØØ4Ø1                          BMI     .+4
        ØØ5116  1Ø44ØØ                  ASL3:   HLT

        ØØ512Ø  ØØØ277                          SCC
        ØØ5122  ØØ5354                          DEC     @-(R4)          ;(RØ)=Ø77777, CC=ØØ1Ø
        ØØ5124  1Ø3ØØ3                          BCC     DEC5
        ØØ5126  1Ø2ØØ2                          BVC     DEC5
        ØØ513Ø  ØØ14Ø1                          BEQ     DEC5
        ØØ5132  1ØØØØ1                          BPL     .+4
        ØØ5134  1Ø44ØØ                  DEC5:   HLT

        ØØ5136  ØØ5453                          NEG     @-(R3)          ;(RØ)=1ØØØØ1, CC=1ØØ1
        ØØ514Ø  1Ø3ØØ2                          BCC     NEG5
        ØØ5142  1Ø24Ø1                          BVS     NEG5
        ØØ5144  1ØØ4Ø1                          BMI     .+4
        ØØ5146  1Ø44ØØ                  NEG5:   HLT

        ØØ515Ø  ØØØ262                          SEV
        ØØ5152  ØØ5134                          COM     @(R4)+          ;(RØ)=Ø77776, CC=ØØØ1
        ØØ5154  1Ø3ØØ1                          BCC     COM3
        ØØ5156  1Ø2ØØ1                          BVC     .+4
        ØØ516Ø  1Ø44ØØ                  COM3:   HLT

        ØØ5162  ØØ5233                          INC     @(R3)+          ;(RØ)=Ø77777, CC=ØØØ1
        ØØ5164  1Ø3ØØ1                          BCC     INC3
        ØØ5166  1ØØØØ1                          BPL     .+4
        ØØ517Ø  1Ø44ØØ                  INC3:   HLT

        ØØ5172  ØØ5554                          ADC     @-(R4)          ;(RØ)=1ØØØØØ, CC=1Ø1Ø
        ØØ5174  1Ø34Ø2                          BCS     ADC5
        ØØ5176  1Ø2ØØ1                          BVC     ADC5
        ØØ52ØØ  1ØØ4Ø1                          BMI     .+4
        ØØ52Ø2  1Ø44ØØ                  ADC5:   HLT

        ØØ52Ø4  ØØØ257                          CCC
        ØØ52Ø6  ØØ6134                          ROL     @(R4)+          ;(RØ)=ØØØØØØ,CC=Ø111
        ØØ521Ø  1Ø3ØØ2                          BCC     ROL3
        ØØ5212  1Ø2ØØ1                          BVC     ROL3
        ØØ5214  ØØ14Ø1                          BEQ     .+4
        ØØ5216  1Ø44ØØ                  ROL3:   HLT

        ØØ522Ø  ØØ5253                          INC     @-(R3)          ;(RØ)=ØØØØØ1, CC=ØØØ1
        ØØ5222  ØØ5654                          SBC     @-(R4)          ;(RØ)=ØØØØØØ, CC=Ø1ØØ
        ØØ5224  1Ø34Ø1                          BCS     SBC5
        ØØ5226  ØØ14Ø1                          BEQ     .+4
        ØØ523Ø  1Ø44ØØ                  SBC5:   HLT
        ØØ5232  1Ø4ØØØ                          SCOPE

                                        ;CHECK UNARY BYTE OPS USING ADDRESS MODES 3 AND 5
        ØØ5234  ØØØ4Ø3                          BR      .+1Ø            ;RESERVE 3 WORDS
        ØØ5236  ØØØØØØ                          .WORD   Ø               ;1 FOR EVEN BYTE ADDRESS
        ØØ524Ø  ØØØØØØ                          .WORD   Ø               ;1 FOR ODD BYTE ADDRESS
        ØØ5242  ØØØØØØ                          .WORD   Ø               ;AND 1 FOR DATA
        ØØ5244  Ø1Ø7Ø2                          MOV     PC,R2
        ØØ5246  ØØ5742                          TST     -(R2)           ;BACK R2 UP TO
        ØØ525Ø  ØØ5742                          TST     -(R2)           ;DATA WORD
        ØØ5252  Ø1Ø2ØØ                          MOV     R2,RØ           ;RØ POINTS TO THE DATA WORD
        ØØ5254  ØØ5Ø1Ø                          CLR     (RØ)            ;PRESET DATA
        ØØ5256  ØØ5742                          TST     -(R2)           ;BACK R2 UP TO
        ØØ526Ø  ØØ5742                          TST     -(R2)           ;EVEN BYTE ADDRESS WORD
        ØØ5262  Ø1ØØ22                          MOV     RØ,(R2)+        ;LOAD ADDRESS
        ØØ5264  ØØ52ØØ                          INC     RØ              ;ODD BYTE ADDRESS
        ØØ5266  Ø1ØØ22                          MOV     RØ,(R2)+        ;LOAD ODD BYTE ADDRESS
        ØØ527Ø  Ø1Ø2ØØ                          MOV     R2,RØ           ;RESET RØ
        ØØ5272  Ø1Ø2Ø5                          MOV     R2,R5

        ØØ5274  1Ø5152                          COMB    @-(R2)          ;(RØ)=1774ØØ,CC=1ØØ1
        ØØ5276  1Ø3ØØ1                          BCC     COMB5
        ØØ53ØØ  1ØØ4Ø1                          BMI     .+4
        ØØ53Ø2  1Ø44ØØ                  COMB5:  HLT

        ØØ53Ø4  1Ø5752                          TSTB    @-(R2)          ;(RØ)=1774ØØ, CC=Ø1ØØ
        ØØ53Ø6  ØØ14Ø1                          BEQ     .+4
```

```
        ØØ531Ø  1Ø44ØØ                          HLT

        ØØ5312  ØØØ262                          SEV
        ØØ5314  1Ø6255                          ASRB    @-(R5)          ;(RØ)=1774ØØ, CC=1ØØ1
        ØØ5316  1Ø3ØØ2                          BCC     ASRB5
        ØØ532Ø  1Ø24Ø1                          BVS     ASRB5
        ØØ5322  1ØØ4Ø1                          BMI     .+4
        ØØ5324  1Ø44ØØ                  ASRB5:  HLT

        ØØ5326  1Ø5232                          INCB    @(R2)+          ;(RØ)=1774Ø1, CC=ØØØ
        ØØ533Ø  1Ø3ØØ1                          BCC     INCB3
        ØØ5332  1ØØØØ1                          BPL     .+4
        ØØ5334  1Ø44ØØ                  INCB3:  HLT

        ØØ5336  ØØØ241                          CLC
        ØØ534Ø  1Ø6Ø55                          RORB    @-(R5)          ;(RØ)=1774ØØ, CC=Ø111
        ØØ5342  1Ø3ØØ3                          BCC     RORB5
        ØØ5344  1Ø2ØØ2                          BVC     RORB5
        ØØ5346  ØØ1ØØ1                          BNE     RORB5
        ØØ535Ø  1ØØØØ1                          BPL     .+4
        ØØ5352  1Ø44ØØ                  RORB5:  HLT

        ØØ5354  1Ø6332                          ASLB    @(R2)+          ;(RØ)=177ØØØ, CC=1ØØ1
        ØØ5356  1Ø3ØØ2                          BCC     ASLB3
        ØØ536Ø  1Ø24Ø1                          BVS     ASLB3
        ØØ5362  1ØØ4Ø1                          BMI     .+4
        ØØ5364  1Ø44ØØ                  ASLB3:  HLT

        ØØ5366  1Ø5552                          ADCB    @-(R2)          ;(RØ)=1774ØØ, CC=1ØØØ
        ØØ537Ø  1Ø34Ø1                          BCS     ADCB5
        ØØ5372  1ØØ4Ø1                          BMI     .+4
        ØØ5374  1Ø44ØØ                  ADCB5:  HLT

        ØØ5376  ØØØ277                          SCC
        ØØ54ØØ  1Ø6135                          ROLB    @(R5)+          ;(RØ)=1774Ø1, CC=ØØØØ
        ØØ54Ø2  1Ø14Ø2                          BLOS    ROLB3           ;BRANCH IF C OR Z IS SET
        ØØ54Ø4  1Ø24Ø1                          BVS     ROLB3
        ØØ54Ø6  1ØØØØ1                          BPL     .+4
        ØØ541Ø  1Ø44ØØ                  ROLB3:  HLT

        ØØ5412  ØØØ352                          SWAB    @-(R2)          ;(RØ)=ØØØ777, CC=1ØØØ
        ØØ5414  1ØØ4Ø1                          BMI     .+4
        ØØ5416  1Ø44ØØ                          HLT

        ØØ542Ø  ØØØ261                          SEC
        ØØ5422  1Ø5635                          SBCB    @(R5)+          ;(RØ)=ØØØ377, CC=Ø1ØØ
        ØØ5424  1Ø34Ø1                          BCS     SBCB3
        ØØ5426  ØØ14Ø1                          BEQ     .+4
        ØØ543Ø  1Ø44ØØ                  SBCB3:  HLT

        ØØ5432  1Ø5432                          NEGB    @(R2)+          ;(RØ)=ØØØØØ1
        ØØ5434  1Ø5352                          DECB    @-(R2)          ;(RØ)=ØØØØØØ, CC=Ø1Ø1
        ØØ5436  1Ø3ØØ1                          BCC     DECB5
        ØØ544Ø  ØØ14Ø1                          BEQ     .+4
        ØØ5442  1Ø44ØØ                  DECB5:  HLT
        ØØ5444  1Ø4ØØØ                          SCOPE

                                        ;CHECK UNARY WORD OPS USING ADDRESS MODE 6 (PC)
        ØØ5446  ØØ5Ø27                          CLR     (PC)+                   ;PRESET DATA = Ø
        ØØ545Ø  ØØØØØØ                  UWM6:   .WORD   Ø                       ;RESERVED FOR DATA
        ØØ5452  Ø1Ø7ØØ                          MOV     PC,RØ
        ØØ5454  Ø24Ø4Ø                          CMP     -(RØ),-(RØ)             ;RØ POINTS TO DATA WORD
        ØØ5456  ØØØ277                          SCC
        ØØ546Ø  ØØ6167  177764                  ROL     UWM6                    ;(RØ)=ØØØØØ1,CC=ØØØØ
        ØØ5464  1Ø34Ø3                          BCS     ROL6
        ØØ5466  1Ø24Ø2                          BVS     ROL6
        ØØ547Ø  ØØ14Ø1                          BEQ     ROL6
        ØØ5472  1ØØØØ1                          BPL     .+4
        ØØ5474  1Ø44ØØ                  ROL6:   HLT

        ØØ5476  ØØ5167  177746                  COM     UWM6                    ;(RØ)=177776, CC=1ØØ1
        ØØ55Ø2  1Ø3ØØ2                          BCC     COM6
        ØØ55Ø4  1Ø24Ø1                          BVS     COM6
        ØØ55Ø6  1ØØ4Ø1                          BMI     .+4
        ØØ551Ø  1Ø44ØØ                  COM6:   HLT
        005512  006267  177732                  ASR     UWM6            ;(R0)=177777, CC=1010
        005516  103402                          BCS     ASR6
        005520  102001                          BVC     ASR6
        005522  100401                          BMI     .+4
        005524  104400                  ASR6:   HLT

        005526  000277                          SCC
        005530  005467  177714                  NEG     UWM6            ;(R0)=000001, CC=0001
        005534  103003                          BCC     NEG6
        005536  102402                          BVS     NEG6
        005540  001401                          BEQ     NEG6
        005542  100001                          BPL     .+4
        005544  104400                  NEG6:   HLT

        005546  000277                          SCC
        005550  006067  177674                  ROR     UWM6            ;(R0)=100000, CC=1001
        005554  103003                          BCC     ROR6
        005556  102402                          BVS     ROR6
        005560  001401                          BEQ     ROR6
        005562  100401                          BMI     .+4
        005564  104400                  ROR6:   HLT

        005566  005667  177656                  SBC     UWM6            ;(R0)=077777, CC=0010
        005572  103402                          BCS     SBC6
        005574  102001                          BVC     SBC6
        005576  100001                          BPL     .+4
        005600  104400                  SBC6:   HLT

        005602  000242                          CLV
        005604  005267  177640                  INC     UWM6            ;(R0)=100000, CC=1011
        005610  103403                          BCS     INC6
        005612  102002                          BVC     INC6
        005614  001401                          BEQ     INC6
        005616  100401                          BMI     .+4
        005620  104400                  INC6:   HLT

        005622  006267  177622                  ASR     UWM6            ;(R0)=140000, CC=1010
        005626  000261                          SEC
        005630  006367  177614                  ASL     UWM6            ;(R0)=100000, CC=1001
        005634  103002                          BCC     ASL6
        005636  102401                          BVS     ASL6
        005640  100401                          BMI     .+4
        005642  104400                  ASL6:   HLT

        005644  005367  177600                  DEC     UWM6            ;(R0)=077777, CC=0011
        005650  103002                          BCC     DEC6
        005652  102001                          BVC     DEC6
        005654  100001                          BPL     .+4
        005656  104400                  DEC6:   HLT

        005660  005567  177564                  ADC     UWM6            ;(R0)=100000, CC=1010
        005664  103402                          BCS     ADC6
```

```
        005666  102001                          BVC     ADC6
        005670  100401                          BMI     .+4
        005672  104400                  ADC6:   HLT
        005674  000242                          CLV
        005676  000367  177546                  SWAB    UWM6
        005702  100401                          BMI     .+4
        005704  104400                          HLT
        005706  022710  000200                  CMP     #200,(R0)
        005712  001401                          BEQ     .+4
        005714  104400                          HLT
        005716  104000                          SCOPE

                                        ;CHECK UNARY BYTE OPS (EVEN/ODD) USING ADDRESS MODE 6 (PC)
        005720  012700  006262                  MOV     #UBM6,R0
        005724  063700  001004                  ADD     @#FACTOR,R0     ;R0 POINTS TO ADDRESS OF DATA
        005730  005067  000326                  CLR     UBM6            ;CLEAR DATA
        005734  000277                          SCC
        005736  000244                          CLZ
        005740  105767  000316                  TSTB    UBM6
        005744  103403                          BCS     TSTB6
        005746  102402                          BVS     TSTB6
        005750  001001                          BNE     TSTB6
        005752  100001                          BPL     .+4
        005754  104400                  TSTB6:  HLT

        005756  000257                          CCC
        005760  105767  000277                  TSTB    UBM6+1          ;TEST ODD BYTE
        005764  001401                          BEQ     .+4
        005766  104400                          HLT

        005770  105667  000266                  SBCB    UBM6            ;(R0)=000000, CC=0100
        005774  103402                          BCS     SBCB6
        005776  102401                          BVS     SBCB6
        006000  001401                          BEQ     .+4
        006002  104400                  SBCB6:  HLT

        006004  000261                  1$:     SEC
        006006  105267  000250                  INCB    UBM6            ;LOOP UNTIL (R0)=077600, CC=1011
        006012  100403                          BMI     2$
        006014  105567  000243                  ADCB    UBM6+1          ;INCB INST INCREMENTS EVEN BYTE
        006020  000771                          BR      1$              ;ADCB INCREMENTS ODD BYTE
        006022  103001                  2$:     BCC     INCB6
        006024  102401                          BVS     .+4
        006026  104400                  INCB6:  HLT

        006030  106367  000226                  ASLB    UBM6            ;(R0)=077400, CC=0111
        006034  103003                          BCC     ASLB6
        006036  102002                          BVC     ASLB6
        006040  001001                          BNE     ASLB6
        006042  100001                          BPL     .+4
        006044  104400                  ASLB6:  HLT

        006046  000242                          CLV
        006050  105567  000207                  ADCB    UBM6+1          ;(R0)=100000, CC=1010
        ØØ6Ø54  1Ø34Ø2                             BCS     ADCB6
        ØØ6Ø56  1Ø2ØØ1                             BVC     ADCB6
        ØØ6Ø6Ø  1ØØ4Ø1                             BMI     .+4
        ØØ6Ø62  1Ø44ØØ                     ADCB6:  HLT

        ØØ6Ø64  ØØØ261                             SEC
        ØØ6Ø66  1Ø6Ø67  ØØØ171                     RORB    UBM6+1          ;(RØ)=14ØØØØ, CC=1Ø1Ø
        ØØ6Ø72  1Ø34Ø2                             BCS     RORB6
        ØØ6Ø74  1Ø2ØØ1                             BVC     RORB6
        ØØ6Ø76  1ØØ4Ø1                             BMI     .+4
        ØØ61ØØ  1Ø44ØØ                     RORB6:  HLT

        ØØ61Ø2  1Ø5167  ØØØ154                     COMB    UBM6            ;(RØ)=14Ø377 CC=1ØØ1
        ØØ61Ø6  1Ø3ØØ2                             BCC     COMB6
        ØØ611Ø  1Ø24Ø1                             BVS     COMB6
        ØØ6112  1ØØ4Ø1                             BMI     .+4
        ØØ6114  1Ø44ØØ                     COMB6:  HLT

        ØØ6116  ØØØ262                             SEV
        ØØ612Ø  1Ø5467  ØØØ137                     NEGB    UBM6+1          ;(RØ)=Ø4Ø377, CC=ØØØ1
        ØØ6124  1Ø3ØØ2                             BCC     NEGB6
        ØØ6126  1Ø24Ø1                             BVS     NEGB6
        ØØ613Ø  1ØØØØ1                             BPL     .+4
        ØØ6132  1Ø44ØØ                     NEGB6:  HLT

        ØØ6134  1Ø6167  ØØØ123                     ROLB    UBM6+1          ;(RØ)=1ØØ777, CC=1Ø1Ø
        ØØ614Ø  1Ø34Ø2                             BCS     ROLB6
        ØØ6142  1Ø2ØØ1                             BVC     ROLB6
        ØØ6144  1ØØ4Ø1                             BMI     .+4
        ØØ6146  1Ø44ØØ                     ROLB6:  HLT

        ØØ615Ø  1Ø6267  ØØØ1Ø6                     ASRB    UBM6            ;(RØ)=1ØØ777, CC=1ØØ1
        ØØ6154  1Ø3ØØ2                             BCC     ASRB6
        ØØ6156  1Ø24Ø1                             BVS     ASRB6
        ØØ616Ø  1ØØ4Ø1                             BMI     .+4
        ØØ6162  1Ø44ØØ                     ASRB6:  HLT

        ØØ6164  1Ø5267  ØØØØ72                     INCB    UBM6            ;(RØ)=1ØØ4ØØ, CC=Ø1Ø1
        ØØ617Ø  1Ø3ØØ2                             BCC     INCB6A
        ØØ6172  1Ø24Ø1                             BVS     INCB6A
        ØØ6174  ØØ14Ø1                             BEQ     .+4
        ØØ6176  1Ø44ØØ                     INCB6A: HLT

        ØØ62ØØ  1Ø5367  ØØØØ57                     DECB    UBM6+1          ;(RØ)=1ØØØØØ, CC=1ØØ1
        ØØ62Ø4  1Ø3ØØ3                             BCC     DECB6A
        ØØ62Ø6  1Ø24Ø2                             BVS     DECB6A
        ØØ621Ø  ØØ14Ø1                             BEQ     DECB6A
        ØØ6212  1ØØ4Ø1                             BMI     .+4
        ØØ6214  1Ø44ØØ                     DECB6A: HLT

        ØØ6216  ØØØ367  ØØØØ4Ø                     SWAB    UBM6            ;(RØ)=ØØØ2ØØ, CC=1ØØØ
        ØØ6222  1Ø34Ø1                             BCS     SWAB6
        ØØ6224  1ØØ4Ø1                             BMI     .+4
        ØØ6226  1Ø44ØØ                     SWAB6:  HLT
```

```
        ØØ623Ø  1Ø6167  ØØØØ26                     ROLB    UBM6            ;(RØ)=ØØØØØØ, CC=Ø111
        ØØ6234  1Ø3ØØ2                             BCC     ROLB6A
        ØØ6236  1Ø2ØØ1                             BVC     ROLB6A
        ØØ624Ø  ØØ14Ø1                             BEQ     .+4
        ØØ6242  1Ø44ØØ                     ROLB6A: HLT

        ØØ6244  ØØ5767  ØØØØ12                     TST     UBM6            ;(RØ)=ØØØØØØ, CC=Ø1ØØ
        ØØ625Ø  1Ø34Ø2                             BCS     TST6
        ØØ6252  1Ø24Ø1                             BVS     TST6
        ØØ6254  ØØ14Ø1                             BEQ     .+4
        ØØ6256  1Ø44ØØ                     TST6:   HLT

        ØØ626Ø  ØØØ4Ø1                             BR      .+4             ;RESERVE A WORD
        ØØ6262  ØØØØØØ                     UBM6:   .WORD   Ø               ;WORD RESERVED FOR DATA
        ØØ6264  1Ø4ØØØ                             SCOPE
        ØØ6266  Ø1Ø7Ø2                             MOV     PC,R2
        ØØ627Ø  Ø627Ø2  ØØØØ12                     ADD     #12,R2
        ØØ6274  Ø127Ø7  ØØ1132                     MOV     #RELOC,PC      ;GO RELOCATE PROGRAM CODE
        ØØ63ØØ  ØØØ24Ø                             NOP                    ;PROGRAM RETURNS HERE+2
                                           ;ØØØØØØØØØØØØØ LAST ADDRESS OF CODE TO BE RELOCATED ØØØØØØØØØØØ
```

```
                                        ;1111111111111 FIRST ADDRESS TO BE RELOCATED 111111111
     ØØ63Ø2  Ø1Ø7ØØ                     REL1:   MOV     PC,RØ               ;GET PC
     ØØ63Ø4  ØØ574Ø                             TST     -(RØ)               ;RØ CONTAINS THE ADDRESS OF REL1
     ØØ63Ø6  Ø1ØØ37  ØØ1Ø1Ø                     MOV     RØ,@#FRSTAD         ;SAVE
     ØØ6312  Ø1Ø7ØØ                             MOV     PC,RØ               ;GET CURRENT PC
     ØØ6314  1627ØØ  ØØ6314                     SUB     #.,RØ               ;SUBTRACT RELOCATION FACTOR
     ØØ632Ø  Ø1ØØ37  ØØ1ØØ4                     MOV     RØ,@#FACTOR         ;SAVE RELOCATION FACTOR
     ØØ6324  Ø1Ø7Ø1                             MOV     PC,R1               ;SET NEW SCOPE PTR

                                        ;CHECK UNARY WORD OPS USING ADDRESS MODE 7
     ØØ6326  ØØØ4Ø3                             BR      UW7                 ;RESERVE 3 WORDS FOR ADDRESSES & DATA
     ØØ633Ø  ØØØØØØ                             .WORD   Ø                   ;CONTAINS ADDRESS OF UWM7
     ØØ6332  ØØØØØØ                     UWM7:   .WORD   Ø                   ;CONTAINS DATA
     ØØ6334  ØØØØØØ                             .WORD   Ø                   ;CONTAINS ADDRESS OF UWM7

     ØØ6336  Ø1Ø7ØØ                     UW7:    MOV     PC,RØ
     ØØ634Ø  ØØ574Ø                             TST     -(RØ)
     ØØ6342  ØØ574Ø                             TST     -(RØ)
     ØØ6344  ØØ5Ø4Ø                             CLR     -(RØ)               ;CLEAR TEST DATA
     ØØ6346  Ø1ØØØ2                             MOV     RØ,R2
     ØØ635Ø  Ø1Ø24Ø                             MOV     R2,-(RØ)            ;SET UP ADDRESS
     ØØ6352  ØØ572Ø                             TST     (RØ)+               ;MOVE RO TO NEXT ADDRESS
     ØØ6354  ØØ572Ø                             TST     (RØ)+
     ØØ6356  Ø1Ø21Ø                             MOV     R2,@(RØ)            ;SET NEXT ADDRESS
     ØØ636Ø  Ø1Ø2ØØ                             MOV     R2,RØ               ;SET RØ POINTING TO DATA
     ØØ6362  ØØØ277                             SCC
     ØØ6364  ØØØ244                             CLZ
     ØØ6366  ØØ5772  ØØØØØ2                     TST     @2(2)               ;(RØ)=ØØØØØØ, CC=Ø1ØØ
     ØØ6372  ØØ14Ø1                             BEQ     .+4
     ØØ6374  1Ø44ØØ                             HLT

     ØØ6376  ØØØ277                             SCC
     ØØ64ØØ  ØØ5672  177776                     SBC     @-2(2)              ;(RØ)=177777, CC=1ØØ1
     ØØ64Ø4  1Ø3ØØ2                             BCC     SBC7
     ØØ64Ø6  1Ø24Ø1                             BVS     SBC7
     ØØ641Ø  1ØØ4Ø1                             BMI     .+4
     ØØ6412  1Ø44ØØ                     SBC7:   HLT

     ØØ6414  ØØØ277                             SCC
     ØØ6416  ØØØ241                             CLC
     ØØ642Ø  ØØ6372  ØØØØØ2                     ASL     @2(2)               ;(RØ)=177776, CC=1ØØ1
     ØØ6424  1Ø3ØØ2                             BCC     ASL7
     ØØ6426  1Ø24Ø1                             BVS     ASL7
     ØØ643Ø  1ØØ4Ø1                             BMI     .+4
     ØØ6432  1Ø44ØØ                     ASL7:   HLT

     ØØ6434  ØØØ257                             CCC
     ØØ6436  ØØ5372  ØØØØØ2                     DEC     @2(2)               ;(RØ)=177775, CC=1ØØØ
     ØØ6442  1Ø34Ø2                             BCS     DEC7
     ØØ6444  1Ø24Ø1                             BVS     DEC7
     ØØ6446  1ØØ4Ø1                             BMI     .+4
     ØØ645Ø  1Ø44ØØ                     DEC7:   HLT
```

```
     ØØ6452  ØØØ262                             SEV
     ØØ6454  ØØ6272  177776                     ASR     @-2(2)              ;(RØ)=177776, CC=1ØØ1
     ØØ646Ø  1Ø3ØØ2                             BCC     ASR7
     ØØ6462  1Ø24Ø1                             BVS     ASR7
     ØØ6464  1ØØ4Ø1                             BMI     .+4
     ØØ6466  1Ø44ØØ                     ASR7:   HLT

     ØØ647Ø  ØØØ241                             CLC
     ØØ6472  ØØØ262                             SEV
     ØØ6474  ØØ6Ø72  177776                     ROR     @-2(2)              ;(RØ)=Ø77777, CC=ØØØØ
     ØØ65ØØ  1Ø14Ø2                             BLOS    ROR7                ;BRANCH IF C OR Z IS SET
     ØØ65Ø2  1Ø24Ø1                             BVS     ROR7
     ØØ65Ø4  1ØØØØ1                             BPL     .+4
     ØØ65Ø6  1Ø44ØØ                     ROR7:   HLT

     ØØ651Ø  ØØØ262                             SEV
     ØØ6512  ØØ5472  ØØØØØ2                     NEG     @2(2)               ;(RØ)=1ØØØØ1, CC=1ØØ1
     ØØ6516  1Ø3ØØ2                             BCC     NEG7
     ØØ652Ø  1Ø24Ø1                             BVS     NEG7
     ØØ6522  1ØØ4Ø1                             BMI     .+4
     ØØ6524  1Ø44ØØ                     NEG7:   HLT

     ØØ6526  ØØØ25Ø                             CLN
     ØØ653Ø  ØØØ372  177776                     SWAB    @-2(2)              ;(RØ)=ØØØ6ØØ, CC=1ØØØ
     ØØ6534  1Ø34Ø1                             BCS     SWAB7
     ØØ6536  1ØØ4Ø1                             BMI     .+4
     ØØ654Ø  1Ø44ØØ                     SWAB7:  HLT

     ØØ6542  ØØØ262                             SEV
     ØØ6544  ØØ5172  ØØØØØ2                     COM     @2(2)               ;(RØ)=177177, CC=1ØØ1
     ØØ655Ø  1Ø3ØØ2                             BCC     COM7
     ØØ6552  1Ø24Ø1                             BVS     COM7
     ØØ6554  1ØØ4Ø1                             BMI     .+4
     ØØ6556  1Ø44ØØ                     COM7:   HLT

     ØØ656Ø  ØØØ372  ØØØØØ2                     SWAB    @2(2)               ;(RØ)=Ø77776, CC=1ØØØ
     ØØ6564  1ØØ4Ø1                             BMI     .+4
     ØØ6566  1Ø44ØØ                             HLT

     ØØ657Ø  ØØØ277                             SCC
     ØØ6572  ØØ5572  177776                     ADC     @-2(2)              ;(RØ)=Ø77777, CC=ØØØØ
     ØØ6576  1Ø34Ø2                             BCS     ADC7
     ØØ66ØØ  1Ø24Ø1                             BVS     ADC7
     ØØ66Ø2  1ØØØØ1                             BPL     .+4
     ØØ66Ø4  1Ø44ØØ                     ADC7:   HLT

     ØØ66Ø6  ØØ5272  ØØØØØ2                     INC     @2(2)               ;(RØ)=1ØØØØØ, CC=1Ø1Ø
     ØØ6612  1Ø2ØØ1                             BVC     INC7
     ØØ6614  1ØØ4Ø1                             BMI     .+4
     ØØ6616  1Ø44ØØ                     INC7:   HLT

     ØØ662Ø  ØØØ257                             CCC
     ØØ6622  ØØ6172  177776                     ROL     @-2(2)              ;(RØ)=ØØØØØØ, CC=Ø111
```

```
        006626  103002                          BCC     ROL7
        006630  102001                          BVC     ROL7
        006632  001401                          BEQ     .+4
        006634  104400                  ROL7:   HLT
        006636  104000                          SCOPE

                                        ;CHECK UNARY BYTE OPS USING ADDRESS MODE 7
        006640  005720                          TST     (R0)+
        006642  005210                          INC     (R0)            ;WORD FOLLOWING UWM7 CONTAINS ADDRESS
        006644  005740                          TST     -(R0)           ;OF ODD BYTE, R0 POINTS TO DATA WORD
        006646  005010                          CLR     (R0)            ;PRESET DATA
        006650  010701                          MOV     PC,R1           ;SET SCOPE PTR
                                        ;NOTE:  @2(2) REFERENCES THE ODD BYTE, AND @-2(2) REFERENCES THE EVEN BYTE,

        006652  000263                          +SEC!SEV                ;SET C AND V
        006654  105672  000002                  SBCB    @2(2)           ;(R0)=177400, CC=1001
        006660  103003                          BCC     SBCB7
        006662  102402                          BVS     SBCB7
        006664  001401                          BEQ     SBCB7
        006666  100401                          BMI     .+4
        006670  104400                  SBCB7:  HLT

        006672  000277                          SCC                     ;SET CONDITION CODES
        006674  105572  177776                  ADCB    @-2(2)          ;(R0)=177401, CC=0000
        006700  103403                          BCS     ADCB7
        006702  102402                          BVS     ADCB7
        006704  001401                          BEQ     ADCB7
        006706  100001                          BPL     .+4
        006710  104400                  ADCB7:  HLT

        006712  105172  177776                  COMB    @-2(2)          ;(R0)=177776, CC=1001
        006716  103002                          BCC     COMB7
        006720  102401                          BVS     COMB7
        006722  100401                          BMI     .+4
        006724  104400                  COMB7:  HLT

        006726  000241                          CLC                     ;CLEAR CARRY
        006730  106072  000002                  RORB    @2(2)           ;(R0)=077776, CC=0011
        006734  103002                          BCC     RORB7
        006736  102001                          BVC     RORB7
        006740  100001                          BPL     .+4
        006742  104400                  RORB7:  HLT

        006744  105272  000002                  INCB    @2(2)           ;(R0)=100376, CC=1011
        006750  103002                          BCC     INCB7
        006752  102001                          BVC     INCB7
        006754  100401                          BMI     .+4
        006756  104400                  INCB7:  HLT

        006760  105372  177776                  DECB    @-2(2)          ;(R0)=100375, CC=1001
        006764  103002                          BCC     DECB7
        006766  102401                          BVS     DECB7
        006770  100401                          BMI     .+4
        006772  104400                  DECB7:  HLT
```

```
        006774  106372  000002                  ASLB    @2(2)           ;(R0)=000375, CC=0111
        007000  103002                          BCC     ASLB7
        007002  102001                          BVC     ASLB7
        007004  001401                          BEQ     .+4
        007006  104400                  ASLB7:  HLT

        007010  000241                          CLC                     ;CLEAR CARRY
        007012  106272  177776                  ASRB    @-2(2)          ;(R0)=000376, CC=1001
        007016  103002                          BCC     ASRB7
        007020  102401                          BVS     ASRB7
        007022  100401                          BMI     .+4
        007024  104400                  ASRB7:  HLT

        007026  105472  000002                  NEGB    @2(2)           ;(R0)=000376, CC=0100
        007032  103402                          BCS     NEGB7
        007034  102401                          BVS     NEGB7
        007036  001401                          BEQ     .+4
        007040  104400                  NEGB7:  HLT

        007042  000262                          SEV
        007044  106172  177776                  ROLB    @-2(2)          ;(R0)=00374, CC=1001
        007050  103002                          BCC     ROLB7
        007052  102401                          BVS     ROLB7
        007054  100401                          BMI     .+4
        007056  104400                  ROLB7:  HLT

        007060  105272  177776                  INCB    @-2(2)          ;(R0)=000375, CC=1001
        007064  105272  177776                  INCB    @-2(2)          ;(R0)=000376, CC=1001
        007070  105572  177776                  ADCB    @-2(2)          ;(R0)=000377, CC=1000
        007074  105172  177776                  COMB    @-2(2)          ;(R0)=000000, CC=0100
        007100  001401                          BEQ     .+4
        007102  104400                          HLT
        007104  104000                          SCOPE

                                        ;CHECK BINARY OPS  USING ADDRESS MODE 0
        007106  000277                          SCC                     ;SET CONDITION CODES
        007110  010700                          MOV     PC,R0           ;R0=PC, CC=X001
        007112  103002                          BCC     MOV0
        007114  102401                          BVS     MOV0
        007116  001001                          BNE     .+4
        007120  104400                  MOV0:   HLT

        007122  010002                          MOV     R0,R2           ;R2=R0
        007124  000262                          SEV                     ;SET V
        007126  160002                          SUB     R0,R2           ;R2=000000, CC=0100
        007130  103402                          BCS     SUB0
        007132  102401                          BVS     SUB0
        007134  001401                          BEQ     .+4
        007136  104400                  SUB0:   HLT

        007140  000244                          CLZ
        007142  010203                          MOV     R2,R3           ;R2=R3=000000, CC=0100
        007144  103401                          BCS     MOV0A
        ØØ7146  ØØ14Ø1                          BEQ     .+4
        ØØ715Ø  1Ø44ØØ                  MOVØA:  HLT

        ØØ7152  ØØØ257                          CCC
        ØØ7154  ØØØ272                          +SEV!SEN                ;SET V & N
        ØØ7156  Ø2Ø2Ø3                          CMP     R2,R3           ;R2=R3=ØØØØØØ, CC=Ø1ØØ
        ØØ716Ø  1Ø34Ø3                          BCS     CMPØ
        ØØ7162  1Ø24Ø2                          BVS     CMPØ
        ØØ7164  ØØ1ØØ1                          BNE     CMPØ
        ØØ7166  1ØØØØ1                          BPL     .+4
        ØØ717Ø  1Ø44ØØ                  CMPØ:   HLT

        ØØ7172  Ø1ØØØ2                          MOV     RØ,R2           ;RØ=R2
        ØØ7174  Ø1Ø2Ø3                          MOV     R2,R3           ;RØ=R2=R3
        ØØ7176  Ø6Ø2Ø3                          ADD     R2,R3           ;R3=2*RØ
        ØØ72ØØ  ØØ63Ø2                          ASL     R2              ;R2=2*RØ
        ØØ72Ø2  Ø2Ø2Ø3                          CMP     R2,R3           ;R2=R3=2*RØ
        ØØ72Ø4  ØØ14Ø1                          BEQ     .+4
        ØØ72Ø6  1Ø44ØØ                          HLT                     ;ERROR; CHECK ADD INSTRUCTION

                                        ;THE FOLLOWING SUBTEST SHIFTS A BIT THROUGH R2 AND R5 AND DOES A
                                        ;BIT TEST (BIT) USING R2 AND R5.
        ØØ721Ø  ØØ5ØØ2                          CLR     R2
        ØØ7212  ØØ52Ø2                          INC     R2
        ØØ7214  ØØØ4Ø2                          BR      2$
        ØØ7216  ØØ63Ø2                  1$:     ASL     R2
        ØØ722Ø  1ØØ4Ø7                          BMI     4$
        ØØ7222  Ø1Ø2Ø5                  2$:     MOV     R2,R5
        ØØ7224  ØØØ277                          SCC
        ØØ7226  Ø3Ø2Ø5                          BIT     R2,R5           ;R2=R5
        ØØ723Ø  1Ø3ØØ2                          BCC     3$
        ØØ7232  1Ø24Ø1                          BVS     3$
        ØØ7234  ØØ137Ø                          BNE     1$
        ØØ7236  1Ø44ØØ                  3$:     HLT
        ØØ724Ø  Ø1Ø2Ø5                  4$:     MOV     R2,R5
        ØØ7242  ØØØ257                          CCC
        ØØ7244  Ø3Ø2Ø5                          BIT     R2,R5
        ØØ7246  1ØØ4Ø1                          BMI     .+4
        ØØ725Ø  1Ø44ØØ                          HLT

        ØØ7252  ØØ5ØØ2                          CLR     R2
        ØØ7254  ØØØ277                          SCC
        ØØ7256  Ø5ØØØ2                          BIS     RØ,R2
        ØØ726Ø  1Ø3ØØ2                          BCC     BISØ
        ØØ7262  1Ø24Ø1                          BVS     BISØ
        ØØ7264  ØØ1ØØ1                          BNE     .+4
        ØØ7266  1Ø44ØØ                  BISØ:   HLT

        ØØ727Ø  Ø1ØØØ3                          MOV     RØ,R3
        ØØ7272  ØØØ277                          SCC
        ØØ7274  ØØØ244                          CLZ
        ØØ7276  Ø4ØØØ3                          BIC     RØ,R3
        ØØ73ØØ  1Ø3ØØ3                          BCC     BICØ
        ØØ73Ø2  1Ø24Ø2                          BVS     BICØ
```

```
        ØØ73Ø4  ØØ1ØØ1                          BNE     BICØ
        ØØ73Ø6  1ØØØØ1                          BPL     .+4
        ØØ731Ø  1Ø44ØØ                  BICØ:   HLT

        ØØ7312  Ø1ØØØ4                          MOV     RØ,R4
        ØØ7314  ØØ51Ø4                          COM     R4
        ØØ7316  Ø4ØØØ4                          BIC     RØ,R4
        ØØ732Ø  ØØ51Ø4                          COM     R4
        ØØ7322  Ø2ØØØ4                          CMP     RØ,R4
        ØØ7324  ØØ14Ø1                          BEQ     .+4
        ØØ7326  1Ø44ØØ                          HLT

        ØØ733Ø  Ø1ØØØ4                          MOV     RØ,R4
        ØØ7332  ØØ51Ø4                          COM     R4
        ØØ7334  Ø1Ø4Ø3                          MOV     R4,R3
        ØØ7336  Ø5ØØØ3                          BIS     RØ,R3
        ØØ734Ø  1Ø3ØØ1                          BCC     BISØA
        ØØ7342  1ØØ4Ø1                          BMI     .+4
        ØØ7344  1Ø44ØØ                  BISØA:  HLT
        ØØ7346  ØØ52Ø3                          INC     R3
        ØØ735Ø  ØØ14Ø1                          BEQ     .+4
        ØØ7352  1Ø44ØØ                          HLT
        ØØ7354  Ø1Ø3Ø4                          MOV     R3,R4           ;R3=R4=Ø
        ØØ7356  ØØ51Ø3                          COM     R3              ;R3=177777
        ØØ736Ø  ØØØ261                          SEC                     ;SET C
        ØØ7362  ØØ6ØØ4                          ROR     R4              ;R4=1ØØØØØ
        ØØ7364  Ø6Ø3Ø4                          ADD     R3,R4           ;R3=177777,R4=Ø77777, CC=ØØ11
        ØØ7366  1Ø3ØØ3                          BCC     ADDØ
        ØØ737Ø  1Ø2ØØ2                          BVC     ADDØ
        ØØ7372  ØØ14Ø1                          BEQ     ADDØ
        ØØ7374  1ØØØØ1                          BPL     .+4
        ØØ7376  1Ø44ØØ                  ADDØ:   HLT
        ØØ74ØØ  Ø1Ø7ØØ                          MOV     PC,RØ
        ØØ74Ø2  Ø22Ø2Ø                          CMP     (RØ)+,(RØ)+
        ØØ74Ø4  Ø2ØØØ7                          CMP     RØ,PC
        ØØ74Ø6  ØØ14Ø1                          BEQ     .+4
        ØØ741Ø  1Ø44ØØ                          HLT

        ØØ7412  Ø1Ø7ØØ                          MOV     PC,RØ
        ØØ7414  Ø627ØØ  ØØØØ1Ø                  ADD     #1Ø,RØ
        ØØ742Ø  Ø1ØØØ2                          MOV     RØ,R2
        ØØ7422  Ø2Ø7ØØ                          CMP     PC,RØ
        ØØ7424  ØØ1ØØ2                          BNE     CMPØA
        ØØ7426  Ø2Ø2ØØ                          CMP     R2,RØ
        ØØ743Ø  ØØ14Ø1                          BEQ     .+4
        ØØ7432  1Ø44ØØ                  CMPØA:  HLT
        ØØ7434  1Ø4ØØØ                          SCOPE

                                        ;CHECK BINARY BYTE OPS USING ADDRESS MODE Ø.
        ØØ7436  Ø127Ø3  125252                  MOV     #125252,R3
        ØØ7442  Ø1Ø3Ø4                          MOV     R3,R4           ;R3=R4=125252
        ØØ7444  14Ø3Ø4                          BICB    R3,R4           ;R3=125252, R4=125ØØØ
        ØØ7446  Ø227Ø4  125ØØØ                  CMP     #125ØØØ,R4
        ØØ7452  ØØ14Ø1                          BEQ     .+4
        007454  104400                          HLT                     ;ERROR! BICB FAILED

        007456  005004                          CLR     R4              ;R3=125252, R4=0
        007460  150304                          BISB    R3,R4           ;R3=125252, R4=000252
        007462  022704  000252                  CMP     #252,R4
        007466  001401                          BEQ     .+4
        007470  104400                          HLT                     ;ERROR! BISB FAILED

        007472  110404                          MOVB    R4,R4           ;R4=177652
        007474  022704  177652                  CMP     #177652,R4      ;MOVB EXTENDS THE SIGN
        007500  001401                          BEQ     .+4
        007502  104400                          HLT                     ;ERROR! MOVB FAILED

        007504  132704  177525                  BITB    #177525,R4
        007510  001401                          BEQ     .+4
        007512  104400                          HLT                     ;ERROR! BITB FAILED

        007514  105104                          COMB    R4              ;R4=177525
        007516  110404                          MOVB    R4,R4           ;R4=000125
        007520  022704  000125                  CMP     #125,R4
        007524  001401                          BEQ     .+4
        007526  104400                          HLT

        007530  150304                          BISB    R3,R4           ;R3=125252, R4=000377
        007532  105204                          INCB    R4
        007534  005704                          TST     R4
        007536  001401                          BEQ     .+4
        007540  104400                          HLT
        007542  104000                          SCOPE

                                        ;CHECK BINARY OPS USING ADDRESS MODE 1
        007544  000402                          BR      .+6             ;RESERVE TWO WORDS
        007546  000000                          .WORD   0               ;RESERVED FOR SOURCE DATA
        007550  000000                          .WORD   0               ;RESERVED FOR DESTINATION DATA
        007552  010704                          MOV     PC,R4
        007554  005744                          TST     -(R4)
        007556  005044                          CLR     -(R4)           ;R4 POINTS TO DESTINATION DATA
        007560  010403                          MOV     R4,R3
        007562  005043                          CLR     -(R3)           ;R3 POINTS TO SOUCE DATA

        007564  005113                          COM     (R3)            ;(R3)=177777
        007566  005214                          INC     (R4)            ;(R4)=000001
        007570  000262                          SEV                     ;SET V
        007572  061314                          ADD     (R3),(R4)       ;(R3)=177777,(R4)=000000, CC=0101
        007574  103002                          BCC     ADD1
        007576  102401                          BVS     ADD1
        007600  001401                          BEQ     .+4
        007602  104400                  ADD1:   HLT

        007604  000277                          SCC
        007606  000250                          CLN
        007610  021314                          CMP     (R3),(R4)       ;(R3)=177777,(R4)=000000, CC=1000
        007612  103403                          BCS     CMP1
```

```
        007614  102402                          BVS     CMP1
        007616  001401                          BEQ     CMP1
        007620  100401                          BMI     .+4
        007622  104400                  CMP1:   HLT

        007624  000277                          SCC
        007626  000244                          CLZ
        007630  031314                          BIT     (R3),(R4)       ;(R3)=177777,(R4)=000000, CC=0101
        007632  103002                          BCC     BIT1
        007634  102401                          BVS     BIT1
        007636  001401                          BEQ     .+4
        007640  104400                  BIT1:   HLT

        007642  000277                          SCC
        007644  000245                          +CLC!CLZ
        007646  005114                          COM     (R4)            ;(R4)=177777
        007650  161314                          SUB     (R3),(R4)       ;(R3)=177777,(R4)=000000, CC=0100
        007652  103402                          BCS     SUB1
        007654  102401                          BVS     SUB1
        007656  001401                          BEQ     .+4
        007660  104400                  SUB1:   HLT

        007662  105013                          CLRB    (R3)            ;(R3)=177400
        007664  000313                          SWAB    (R3)            ;(R3)=000377
        007666  000270                          SEN
        007670  011314                          MOV     (R3),(R4)       ;(R3)=(R4)=000377
        007672  100001                          BPL     .+4
        007674  104400                          HLT
        007676  000314                          SWAB    (R4)            ;(R3)=000377,(R4)=177400
        007700  000263                          +SEC!SEV                ;SET C & V
        007702  051314                          BIS     (R3),(R4)       ;(R3)=000377,(R4)=177777, CC=1001
        007704  103002                          BCC     BIS1
        007706  102401                          BVS     BIS1
        007710  100401                          BMI     .+4
        007712  104400                  BIS1:   HLT

        007714  041314                          BIC     (R3),(R4)       ;(R3)=000377,(R4)=177400, CC=1001
        007716  103002                          BCC     BIC1
        007720  102401                          BVS     BIC1
        007722  100401                          BMI     .+4
        007724  104400                  BIC1:   HLT

        007726  000262                          SEV                     ;SET V
        007730  021314                          CMP     (R3),(R4)       ;(R3)=000377,(R4)=177400, CC=0001
        007732  103003                          BCC     CMP1A
        007734  102402                          BVS     CMP1A
        007736  001401                          BEQ     CMP1A
        007740  100001                          BPL     .+4
        007742  104400                  CMP1A:  HLT

        007744  005013                          CLR     (R3)            ;(R3)=000000
        007746  000261                          SEC
        007750  006013                          ROR     (R3)            ;(R3)=100000
        007752  011314                          MOV     (R3),(R4)       ;(R3)=(R4)=100000
        007754  005114                                  COM     (R4)            ;(R4)=077777
        007756  161314                                  SUB     (R3),(R4)       ;(R3)=100000,(R4)=177777, CC=1011
        007760  103002                                  BCC     SUB1A
        007762  102001                                  BVC     SUB1A
        007764  100401                                  BMI     .+4
        007766  104400                          SUB1A:  HLT

        007770  000277                                  SCC
        007772  161314                                  SUB     (R3),(R4)       ;(R3)=100000,(R4)=077777, CC=0000
        007774  101402                                  BLOS    SUB1B           ;BRANCH IF C OR Z IS SET
        007776  102401                                  BVS     SUB1B
        010000  100001                                  BPL     .+4
        010002  104400                          SUB1B:  HLT

        010004  011314                                  MOV     (R3),(R4)       ;(R3)=100000,(R4)=100000, CC=1000
        010006  001401                                  BEQ     MOV1
        010010  100401                                  BMI     .+4
        010012  104400                          MOV1:   HLT

        010014  061314                                  ADD     (R3),(R4)       ;(R3)=100000,(R4)=000000, CC=0111
        010016  103003                                  BCC     ADD1A
        010020  102002                                  BVC     ADD1A
        010022  001001                                  BNE     ADD1A
        010024  100001                                  BPL     .+4
        010026  104400                          ADD1A:  HLT

        010030  005113                                  COM     (R3)            ;(R3)=077777
        010032  011314                                  MOV     (R3),(R4)            ;(R4)=077777
        010034  061314                                  ADD     (R3),(R4)       ;(R3)=077777,(R4)=177776, CC=1010
        010036  103402                                  BCS     ADD1B
        010040  102001                                  BVC     ADD1B
        010042  100401                                  BMI     .+4
        010044  104400                          ADD1B:  HLT

        010046  062714  000002                          ADD     #2,(R4)
        010052  005714                                  TST     (R4)            ;CHECK FINAL RESULT
        010054  001401                                  BEQ     .+4
        010056  104400                                  HLT
        010060  104000                                  SCOPE

                                                ;CHECK BINARY BYTE OPS USING ADDRESS MODE 1
        010062  000402                                  BR      .+6
        010064  000000                                  .WORD   0
        010066  000000                                  .WORD   0
        010070  010705                                  MOV     PC,R5
        010072  005745                                  TST     -(R5)
        010074  005045                                  CLR     -(R5)           ;(R5)=000000
        010076  010502                                  MOV     R5,R2
        010100  005042                                  CLR     -(R2)           ;(R2)=000000
        010102  005202                                  INC     R2              ;R2 POINTS TO ODD BYTE
        010104  105112                                  COMB    (R2)            ;(R2)=177400

        010106  000277                                  SCC
        010110  111215                                  MOVB    (R2),(R5)       ;(R2)=177400,(R5)=000377,CC=1001
```

```
        010112  103005                                  BCC     MOVB1
        010114  102404                                  BVS     MOVB1
        010116  001403                                  BEQ     MOVB1
        010120  100002                                  BPL     MOVB1
        010122  105215                                  INCB    (R5)            ;CHECK RESULT
        010124  001401                                  BEQ     .+4
        010126  104400                          MOVB1:  HLT

        010130  106312                                  ASLB    (R2)            ;SHIFT (R2) UNTIL
        010132  102376                                  BVC     .-2             ;(R2)=000000
        010134  106012                                  RORB    (R2)            ;(R2)=100000
        010136  105315                                  DECB    (R5)            ;(R5)=00377
        010140  106015                                  RORB    (R5)            ;(R5)=000177
        010142  000257                                  CCC
        010144  121512                                  CMPB    (R5),(R2)       ;(R5)=000177,(R2)=100000, CC=1010
        010146  102001                                  BVC     CMPB1
        010150  100401                                  BMI     .+4
        010152  104400                          CMPB1:  HLT

        010154  005003                                  CLR     R3
        010156  000261                                  SEC
        010160  006003                                  ROR     R3              ;R3=100000
        010162  050315                                  BIS     R3,(R5)         ;(R5)=100177
        010164  000273                                  +SEC!SEV!SEN            ;SET C,V, & N
        010166  131215                                  BITB    (R2),(R5)       ;(R2)=100000,(R5)=100177, CC=0101
        010170  103002                                  BCC     BITB1
        010172  102401                                  BVS     BITB1
        010174  001401                                  BEQ     .+4
        010176  104400                          BITB1:  HLT

        010200  151215                                  BISB    (R2),(R5)       ;(R2)=100000,(R5)=100377, CC=1001
        010202  103001                                  BCC     BISB1
        010204  100401                                  BMI     .+4
        010206  104400                          BISB1:  HLT

        010210  141215                                  BICB    (R2),(R5)       ;(R2)=100000,(R5)=100177, CC=0001
        010212  103002                                  BCC     BICB1
        010214  001401                                  BEQ     BICB1
        010216  100001                                  BPL     .+4
        010220  104400                          BICB1:  HLT

        010222  105112                                  COMB    (R2)            ;(R2)=077400,(R5)=100177
        010224  121215                                  CMPB    (R2),(R5)
        010226  001401                                  BEQ     .+4
        010230  104400                                  HLT

        010232  141512                                  BICB    (R5),(R2)       ;(R5)=100177,(R2)=000000, CC=0100
        010234  001002                                  BNE     BICB1A
        010236  105712                                  TSTB    (R2)
        010240  001401                                  BEQ     .+4
        010242  104400                          BICB1A: HLT

        010244  000402                                  BR      .+6             ;RESERVE TWO WORDS FOR DATA
        010246  000000                                  .WORD   0               ;SOURCE DATA
        010250  000000                          .WORD   0               ;DEST DATA
        010252  010705                          MOV     PC,R5
        010254  005745                          TST     -(R5)
        010256  105045                          CLRB    -(R5)           ;R5 POINTS TO DEST ODD BYTE
        010260  010504                          MOV     R5,R4
        010262  105044                          CLRB    -(R4)           ;R4 POINTS TO DEST EVEN BYTE
        010264  010403                          MOV     R4,R3
        010266  105043                          CLRB    -(R3)           ;R3 POINTS TO SOURCE ODD BYTE
        010270  010302                          MOV     R3,R2
        010272  105042                          CLRB    -(R2)           ;R2 POINTS TO SOURCE EVEN BYTE

                                        ;COMMENTS ARE LEAST SIGNIFICANT 4 BITS OF BYTES POINTED TO BY R2,R3
                                        ;R4, AND R5 RESPECTIVELY AND THE REMAINING BITS ARE 0'S.
        010274  000261                          SEC                     ;SET CARRY
                                                                        ;(R2),(R3),(R4),(R5)
        010276  106112                          ROLB    (R2)            ;0001,0000,0000,0000
        010300  111214                          MOVB    (R2),(R4)       ;0001,0000,0001,0000
        010302  106112                          ROLB    (R2)            ;0010,0000,0001,0000
        010304  111213                          MOVB    (R2),(R5)       ;0010,0010,0001,0000
        010306  106112                          ROLB    (R2)            ;0100,0010,0001,0000
        010310  111315                          MOVB    (R3),(R5)       ;0100,0010,0001,0010
        010312  106112                          ROLB    (R2)            ;1000,0010,0001,0010
        010314  106113                          ROLB    (R3)            ;1000,0100,0001,0010
        010316  151215                          BISB    (R2),(R5)       ;1000,0100,0001,1010
        010320  131512                          BITB    (R5),(R2)       ;1000,0100,0001,1010
        010322  001426                          BEQ     BIN1
        010324  151314                          BISB    (R3),(R4)       ;1000,0100,0101,1010
        010326  131413                          BITB    (R4),(R3)       ;1000,0100,0101,1010
        010330  001423                          BEQ     BIN1
        010332  105213                          INCB    (R3)            ;1000,0101,0101,1010
        010334  121314                          CMPB    (R3),(R4)       ;1000,0101,0101,1010
        010336  001020                          BNE     BIN1
        010340  106113                          ROLB    (R3)            ;1000,1010,0101,1010
        010342  121315                          CMPB    (R3),(R5)       ;1000,1010,0101,1010
        010344  001015                          BNE     BIN1
        010346  106212                          ASRB    (R2)            ;0100,1010,0101,1010
        010350  131214                          BITB    (R2),(R4)       ;0100,1010,0101,1010
        010352  001412                          BEQ     BIN1
        010354  106015                          RORB    (R5)            ;0100,1010,0101,0101
        010356  121415                          CMPB    (R4),(R5)       ;0100,1010,0101,0101
        010360  001007                          BNE     BIN1
        010362  105314                          DECB    (R4)            ;0100,1010,0100,0101
        010364  141214                          BICB    (R2),(R4)       ;0100,1010,0000,0101
        010366  001004                          BNE     BIN1
        010370  111314                          MOVB    (R3),(R4)       ;0100,1010,1010,0101
        010372  106213                          ASRB    (R3)            ;0100,0101,1010,0101
        010374  141315                          BICB    (R3),(R5)       ;0100,0101,1010,0101
        010376  001401                          BEQ     .+4
        010400  104400                  BIN1:   HLT
        010402  104000                          SCOPE

                                        ;CHECK BINARY WORD OPS USING ADDRESS MODE 2 & 4
        010404  010405                          MOV     R4,R5           ;SET DESTINATION REGISTER
        010406  012715  000001                  MOV     #1,(R5)
```

```
        010412  012712  177777                  MOV     #-1,(R2)
        010416  000257                          CCC
        010420  000262                          SEV
        010422  062225                          ADD     (R2)+,(R5)+     ;(R2)=177777,(R5)=000000, CC=0101
        010424  103002                          BCC     ADD2
        010426  102401                          BVS     ADD2
        010430  001401                          BEQ     .+4
        010432  104400                  ADD2:   HLT

        010434  000262                          SEV                     ;SET V
        010436  024527  000001                  CMP     -(R5),#1        ;(R5)=000000, CC=1001
        010442  103002                          BCC     CMP2
        010444  102401                          BVS     CMP2
        010446  100401                          BMI     .+4
        010450  104400                  CMP2:   HLT

        010452  054225                          BIS     -(R2),(R5)+     ;(R2)=177777,(R5)=177777,CC=1001
        010454  103001                          BCC     BIS2
        010456  100401                          BMI     .+4
        010460  104400                  BIS2:   HLT
        010462  000277                          SCC
        010464  000244                          CLZ
        010466  162245                          SUB     (R2)+,-(R5)     ;(R2)=177777,(R5)=000000, CC=0100
        010470  103402                          BCS     SUB2
        010472  102401                          BVS     SUB2
        010474  001401                          BEQ     .+4
        010476  104400                  SUB2:   HLT

        010500  005442                          NEG     -(R2)           ;(R2)=000001
        010502  005115                          COM     (R5)            ;(R5)=177777
        010504  000277                          SCC
        010506  000250                          CLN
        010510  042225                          BIC     (R2)+,(R5)+     ;(R2)=000001,(R5)=177776, CC=1001
        010512  103003                          BCC     BIC2
        010514  102402                          BVS     BIC2
        010516  001401                          BEQ     BIC2
        010520  100401                          BMI     .+4
        010522  104400                  BIC2:   HLT

        010524  012742  125252                  MOV     #125252,-(R2)
        010530  012245                          MOV     (R2)+,-(R5)
        010532  005125                          COM     (R5)+           ;(R5)=052525
        010534  000262                          SEV
        010536  034245                          BIT     -(R2),-(R5)     ;(R2)=125252,(R5)=052525, CC=0101
        010540  103002                          BCC     BIT2
        010542  102401                          BVS     BIT2
        010544  001401                          BEQ     .+4
        010546  104400                  BIT2:   HLT

        010550  000262                          SEV
        010552  052225                          BIS     (R2)+,(R5)+     ;(R2)=125252,(R5)=177777, CC=1001
        010554  103002                          BCC     BIS2A
        010556  102401                          BVS     BIS2A
        010560  100401                          BMI     .+4
        Ø1Ø562  1Ø44ØØ                  BIS2A:  HLT

        Ø1Ø564  Ø42745  125252                  BIC     #125252,-(R5)   ;(R5)=Ø52525
        Ø1Ø57Ø  ØØ5125                          COM     (R5)+           ;(R5)=125252
        Ø1Ø572  Ø24245                          CMP     -(R2),-(R5)
        Ø1Ø574  ØØ14Ø1                          BEQ     .+4
        Ø1Ø576  1Ø44ØØ                          HLT

        Ø1Ø6ØØ  ØØ5Ø12                          CLR     (R2)
        Ø1Ø6Ø2  ØØ5122                          COM     (R2)+           ;(R2)=177777
        Ø1Ø6Ø4  162742  ØØØØØ1                  SUB     #1,-(R2)        ;(R2)=177776, CC=1ØØØ
        Ø1Ø61Ø  1Ø34Ø2                          BCS     SUB2A
        Ø1Ø612  1Ø24Ø1                          BVS     SUB2A
        Ø1Ø614  1ØØ4Ø1                          BMI     .+4
        Ø1Ø616  1Ø44ØØ                  SUB2A:  HLT
        Ø1Ø62Ø  1Ø4ØØØ                          SCOPE

        Ø1Ø622  Ø1Ø7Ø2                          MOV     PC,R2           ;GET CURRENT PC
        Ø1Ø624  Ø1Ø2Ø5                          MOV     R2,R5           ;MOVE TO R5
        Ø1Ø626  124245                  1$:     CMPB    -(R2),-(R5)     ;COMPARE ALL PREVIOUS MEMORY ADDRESSES
        Ø1Ø63Ø  ØØ14Ø1                          BEQ     .+4
        Ø1Ø632  1Ø44ØØ                          HLT                     ;ERROR!
        Ø1Ø634  Ø2Ø237  ØØ1Ø1Ø                  CMP     R2,@#FRSTAD     ;CHECK FOR LOW LIMIT
        Ø1Ø64Ø  ØØ1372                          BNE     1$
        Ø1Ø642  1Ø4ØØØ                          SCOPE

                                        ;CHECK BINARY BYTE OPS USING ADDRESS MODES 2 & 4.
        Ø1Ø644  ØØØ4Ø2                          BR      .+6             ;RESERVE TWO WORDS
        Ø1Ø646  ØØØØØØ                          .WORD   Ø               ;SOURCE DATA
        Ø1Ø65Ø  ØØØØØØ                          .WORD   Ø               ;DESTINATION DATA
        Ø1Ø652  Ø1Ø7Ø3                          MOV     PC,R3
        Ø1Ø654  ØØ5743                          TST     -(R3)
        Ø1Ø656  112743  ØØØ2ØØ                  MOVB    #2ØØ,-(R3)
        Ø1Ø662  112743  ØØØ377                  MOVB    #377,-(R3)      ;(R3)=1ØØ377
        Ø1Ø666  Ø1Ø3Ø4                          MOV     R3,R4
        Ø1Ø67Ø  112744  ØØØ177                  MOVB    #177,-(R4)
        Ø1Ø674  112744  ØØØØØØ                  MOVB    #Ø,-(R4)        ;(R4)=Ø774ØØ
        Ø1Ø7ØØ  ØØ14Ø1                          BEQ     .+4
        Ø1Ø7Ø2  1Ø44ØØ                          HLT

        Ø1Ø7Ø4  152324                          BISB    (R3)+,(R4)+     ;(R3)=1ØØ377,(R4)=Ø77777
        Ø1Ø7Ø6  1ØØ4Ø1                          BMI     .+4
        Ø1Ø71Ø  1Ø44ØØ                          HLT

        Ø1Ø712  122324                          CMPB    (R3)+,(R4)+
        Ø1Ø714  1Ø34Ø2                          BCS     CMPB2
        Ø1Ø716  1Ø2ØØ1                          BVC     CMPB2
        Ø1Ø72Ø  1ØØØØ1                          BPL     .+4
        Ø1Ø722  1Ø44ØØ                  CMPB2:  HLT

        Ø1Ø724  ØØØ261                          SEC
        Ø1Ø726  134344                          BITB    -(R3),-(R4)
        Ø1Ø73Ø  1Ø3ØØ2                          BCC     BITB2
        Ø1Ø732  1Ø24Ø1                          BVS     BITB2
```

```
        Ø1Ø734  ØØ14Ø1                          BEQ     .+4
        Ø1Ø736  1Ø44ØØ                  BITB2:  HLT

        Ø1Ø74Ø  ØØØ244                          CLZ
        Ø1Ø742  144344                          BICB    -(R3),-(R4)     ;(R3)=1ØØ377,(R4)=Ø774ØØ
        Ø1Ø744  ØØ14Ø1                          BEQ     .+4
        Ø1Ø746  1Ø44ØØ                          HLT
        Ø1Ø75Ø  1Ø4ØØØ                          SCOPE

                                        ;CHECK BINARY WORD OPS USING ADDRESS MODES 3 & 5.
        Ø1Ø752  ØØØ4Ø4                          BR      2$              ;RESERVE SPACE FOR DATA AND ADDRESSES
        Ø1Ø754  ØØØØØØ                          .WORD   Ø               ;CONTAINS ADDRESS OF SOURCE DATA
        Ø1Ø756  ØØØØØØ                          .WORD   Ø               ;CONTAINS ADDRESS OF DEST DATA
        Ø1Ø76Ø  ØØØØØØ                          .WORD   Ø               ;CONTAINS SOURCE DATA
        Ø1Ø762  ØØØØØØ                          .WORD   Ø               ;CONTAINS DEST DATA
        Ø1Ø764  Ø1Ø7Ø1                  2$:     MOV     PC,R1
        Ø1Ø766  Ø1Ø1ØØ                          MOV     R1,RØ           ;SET SCOPE PTR
        Ø1Ø77Ø  Ø24Ø4Ø                          CMP     -(RØ),-(RØ)     ;ADJUST RØ
        Ø1Ø772  Ø1ØØØ5                          MOV     RØ,R5           ;R5 POINTS TO DEST DATA
        Ø1Ø774  Ø24545                          CMP     -(R5),-(R5)     ;SUB 4 FROM R5
        Ø1Ø776  Ø1ØØ15                          MOV     RØ,(R5)         ;R5 POINTS TO ADDRESS OF DEST DATA
        Ø11ØØØ  Ø1Ø5Ø2                          MOV     R5,R2
        Ø11ØØ2  Ø1ØØØ4                          MOV     RØ,R4           ;R4 POINTS TO DEST DATA
        Ø11ØØ4  ØØ574Ø                          TST     -(RØ)
        Ø11ØØ6  Ø1ØØØ3                          MOV     RØ,R3           ;R3 POINTS TO SOURCE DATA
        Ø11Ø1Ø  Ø1ØØ42                          MOV     RØ,-(R2)        ;R2 POINTS TO ADDRESS OF SOURCE DATA
        Ø11Ø12  ØØ5Ø13                          CLR     (R3)            ;PRESET SOURCE DATA
        Ø11Ø14  ØØ5Ø14                          CLR     (R4)            ;PRESET DEST DATA

        Ø11Ø16  ØØØ277                          SCC
        Ø11Ø2Ø  ØØØ244                          CLZ
        Ø11Ø22  163235                          SUB     @(R2)+,@(R5)+   ;(R3)=ØØØØØØ,(R4)=ØØØØØØ, CC=Ø1ØØ
        Ø11Ø24  1Ø34Ø2                          BCS     SUB3
        Ø11Ø26  1Ø24Ø1                          BVS     SUB3
        Ø11Ø3Ø  ØØ14Ø1                          BEQ     .+4
        Ø11Ø32  1Ø44ØØ                  SUB3:   HLT

        Ø11Ø34  Ø52752  1ØØØØØ                  BIS     #1ØØØØØ,@-(R2)  ;(R3)=1ØØØØØ
        Ø11Ø4Ø  Ø62755  ØØØØØ1                  ADD     #1,@-(R5)       ;(R4)=ØØØØØ1
        Ø11Ø44  163235                          SUB     @(R2)+,@(R5)+   ;(R3)=1ØØØØØ,(R4)=1ØØØØ1, CC=1Ø11
        Ø11Ø46  1Ø3ØØ2                          BCC     SUB3A
        Ø11Ø5Ø  1Ø2ØØ1                          BVC     SUB3A
        Ø11Ø52  1ØØ4Ø1                          BMI     .+4
        Ø11Ø54  1Ø44ØØ                  SUB3A:  HLT

        Ø11Ø56  ØØ5414                          NEG     (R4)            ;(R4)=Ø77777
        Ø11Ø6Ø  Ø35255                          BIT     @-(R2),@-(R5)   ;(R3)=1ØØØØØ,(R4)=Ø77777
        Ø11Ø62  ØØ14Ø1                          BEQ     .+4
        Ø11Ø64  1Ø44ØØ                          HLT
        Ø11Ø66  Ø23235                          CMP     @(R2)+,@(R5)+
        Ø11Ø7Ø  1Ø24Ø1                          BVS     .+4
        Ø11Ø72  1Ø44ØØ                          HLT
        Ø11Ø74  ØØ5152                          COM     @-(R2)
        Ø11Ø76  ØØØ257                          CCC
        011100  063255                                  ADD     @(R2)+,@-(R5)
        011102  102001                                  BVC     ADD3
        011104  100401                                  BMI     .+4
        011106  104400                          ADD3:   HLT
        011110  000261                                  SEC
        011112  045235                                  BIC     @-(R2),@(R5)+   ;(R3)=077777,(R4)=100000
        011114  103001                                  BCC     BIC3
        011116  100401                                  BMI     .+4
        011120  104400                          BIC3:   HLT

        011122  005155                                  COM     @-(R5)          ;(R4)=077777
        011124  023235                                  CMP     @(R2)+,@(R5)+   ;(R3)=077777,(R4)=077777
        011126  001401                                  BEQ     .+4
        011130  104400                                  HLT
        011132  104000                                  SCOPE

                                                ;CHECK BINARY BYTE OPS USING ADDRESS MODES 3 & 5,
        011134  000406                                  BR      1$              ;RESERVE SPACE FOR ADDRESSES & DATA
        011136  000000                                  .WORD   0               ;CONTAINS ADDRESS OF SOURCE DATA (EVEN BYTE)
        011140  000000                                  .WORD   0               ;CONTAINS ADDRESS OF SOURCE DATA (ODD BYTE)
        011142  000000                                  .WORD   0               ;CONTAINS ADDRESS OF DEST DATA (EVEN BYTE)
        011144  000000                                  .WORD   0               ;CONTAINS ADDRESS OF DEST DATA (ODD BYTE)
        011146  000000                                  .WORD   0               ;CONTAINS SOURCE DATA
        011150  000000                                  .WORD   0               ;CONTAINS DEST DATA

        011152  010700                          1$:     MOV     PC,R0
        011154  024040                                  CMP     -(R0),-(R0)     ;R0=ADDRESS OF DEST DATA
        011156  010003                                  MOV     R0,R3           ;R3     "       "
        011160  010305                                  MOV     R3,R5           ;R5     "       "
        011162  005743                                  TST     -(R3)           ;SUB 2 FROM R3
        011164  010043                                  MOV     R0,-(R3)        ;R3 POINTS TO ADDRESS OF DEST DATA
        011166  005213                                  INC     (R3)            ;ODD BYTE
        011170  010043                                  MOV     R0,-(R3)        ;EVEN BYTE
        011172  010304                                  MOV     R3,R4
        011174  005740                                  TST     -(R0)           ;R0=ADDRESS OF SOURCE DATA
        011176  010044                                  MOV     R0,-(R4)        ;R4 POINTS TO ADDRESS OF SOURCE DATA
        011200  005214                                  INC     (R4)            ;ODD BYTE
        011202  010044                                  MOV     R0,-(R4)        ;EVEN BYTE

        011204  000261                                  SEC                     ;SET CARRY
        011206  012734  177001                          MOV     #177001,@(R4)+
        011212  112734  000200                          MOVB    #200,@(R4)+     ;SOURCE DATA=100001
        011216  115433                                  MOVB    @-(R4),@(R3)+
        011220  115433                                  MOVB    @-(R4),@(R3)+   ;DEST DATA=000600
        011222  103401                                  BCS     .+4
        011224  104400                                  HLT                     ;ERROR! MOV DOES AFFECT C BIT IN PSW
        011226  022715  000600                          CMP     #600,(R5)       ;CHECK DEST DATA
        011232  001401                                  BEQ     .+4
        011234  104400                                  HLT                     ;ERROR! INCORRECT RESULT
        011236  024343                                  CMP     -(R3),-(R3)     ;POINT R4 BACK TO EVEN BYTE
        011240  153433                                  BISB    @(R4)+,@(R3)+
        011242  153433                                  BISB    @(R4)+,@(R3)+   ;DEST DATA=100601
        011244  022715  100601                          CMP     #100601,(R5)    ;CHECK RESULT
        011250  001401                                  BEQ     .+4
```

```
        011252  104400                                  HLT                     ;ERROR! INCORRECT DEST DATA AFTER BISB
        011254  145453                                  BICB    @-(R4),@-(R3)
        011256  145453                                  BICB    @-(R4),@-(R3)
        011260  133433                                  BITB    @(R4)+,@(R3)+
        011262  001002                                  BNE     BITB3
        011264  135433                                  BITB    @-(R4),@(R3)+
        011266  001001                                  BNE     .+4
        011270  104400                          BITB3:  HLT

        011272  123453                                  CMPB    @(R4)+,@-(R3)
        011274  001002                                  BNE     CMPB3
        011276  123453                                  CMPB    @(R4)+,@-(R3)
        011300  001401                                  BEQ     .+4
        011302  104400                          CMPB3:  HLT
        011304  104000                                  SCOPE

                                                ;CHECK BINARY OPS USING ADDRESS MODE 6
        011306  000402                                  BR      .+6             ;RESERVE TWO LOCATIONS
        011310  000000                          SDATA:  .WORD   0               ;RESERVED FOR SOURCE DATA
        011312  000000                          DDATA:  .WORD   0               ;RESERVED FOR DESTINATION DATA

        011314  013702  001004                          MOV     @#FACTOR,R2     ;GET RELOCATION FACTOR AND USE AS AN
        011320  010205                                  MOV     R2,R5           ;INDEX VALUE TO POINT TO DATA
        011322  005065  011312                          CLR     DDATA(5)        ;PRESET DESTINATION DATA
        011326  012762  000001  011310                  MOV     #1,SDATA(2)     ;THIS ROUTSINE PUT A 1 BIT INTO EVERY
        011334  056265  011310  011312  1$:     BIS     SDATA(2),DDATA(5)       ;OTHER BIT POSITION IN THE DEST-
        011342  006362  011310                          ASL     SDATA(2)                ;INATION ADDRESS (52525)
        011346  006362  011310                          ASL     SDATA(2)
        011352  103370                                  BCC     1$
        011354  022765  052525  011312                  CMP     #52525,DDATA(5) ;CHECK RESULT
        011362  001401                                  BEQ     .+4
        011364  104400                                  HLT                     ;ERROR! INCORRECT RESULT
        011366  012762  177777  011310                  MOV     #-1,SDATA(2)
        011374  046562  011312  011310                  BIC     DDATA(5),SDATA(2)       ;SOURCE DATA=125252
        011402  036265  011310  011312                  BIT     SDATA(2),DDATA(5)
        011410  001401                                  BEQ     .+4
        011412  104400                                  HLT                     ;ERROR! BIT INST FAILED
        011414  006365  011312                          ASL     DDATA(5)        ;DDATA=125252
        011420  026265  011310  011312                  CMP     SDATA(2),DDATA(5)
        011426  001401                                  BEQ     .+4

        011430  104400                                  HLT                     ;ERROR! CMP INST FAILED
        011432  000257                                  CCC
        011434  066265  011310  011312                  ADD     SDATA(2),DDATA(5)
        011442  103002                                  BCC     ADD6
        011444  102001                                  BVC     ADD6
        011446  100001                                  BPL     .+4
        011450  104400                          ADD6:   HLT

        011452  006362  011310                          ASL     SDATA(2)        ;SDATA=52524
        011456  166265  011310  011312                  SUB     SDATA(2),DDATA(5)
        011464  103401                                  BCS     SUB6
        011466  001401                                  BEQ     .+4
        011470  104400                          SUB6:   HLT
        Ø11472  1127ØØ  ØØØ377                  MOVB    #377,RØ         ;RØ=177777 (MOVB %R EXTENDS SIGN)
        Ø11476  Ø1ØØ62  Ø1131Ø                  MOV     RØ,SDATA(2)
        Ø115Ø2  Ø12765  177777  Ø11312          MOV     #-1,DDATA(5)
        Ø1151Ø  1665ØØ  Ø11312                  SUB     DDATA(5),RØ
        Ø11514  ØØ14Ø1                          BEQ     .+4
        Ø11516  1044ØØ                          HLT
        Ø1152Ø  Ø66265  Ø1131Ø  Ø11312  1$:     ADD     SDATA(2),DDATA(5)
        Ø11526  ØØ6362  Ø1131Ø                  ASL     SDATA(2)
        Ø11532  ØØ5162  Ø1131Ø                  COM     SDATA(2)
        Ø11536  Ø36265  Ø1131Ø  Ø11312          BIT     SDATA(2),DDATA(5)
        Ø11544  ØØ14Ø1                          BEQ     .+4
        Ø11546  1044ØØ                          HLT
        Ø1155Ø  ØØ5162  Ø1131Ø                  COM     SDATA(2)
        Ø11554  Ø26265  Ø1131Ø  Ø11312          CMP     SDATA(2),DDATA(5)
        Ø11562  ØØ14Ø1                          BEQ     .+4
        Ø11564  1044ØØ                          HLT
        Ø11566  Ø262ØØ  Ø1131Ø                  CMP     SDATA(2),RØ
        Ø11572  ØØ1352                          BNE     1$
        Ø11574  1Ø4ØØØ                          SCOPE

                                        ;CHECK BINARY BYTE OPS USING ADDRESS MODE 6
                                        ;NOTE: SDATAB(2), AND DDATAB(4) REFERENCE EVEN BYTE OF SOURCE & DEST DATA
                                        ;AND SDATAB(3), AND DDATAB(5) FEFERENCE ODD BYTE OF SOURCE & DEST DATA

        Ø11576  Ø137Ø2  ØØ1ØØ4                  MOV     @#FACTOR,R2     ;GET INDEX VALUE
        Ø116Ø2  Ø1Ø2Ø4                          MOV     R2,R4           ;R2 FOR SOURCE EVEN BYTE INDEX, R4 FOR
        Ø116Ø4  Ø1Ø4Ø3                          MOV     R4,R3           ;DEST ODD BYTE, R3 FOR SOURCE EVEN
        Ø116Ø6  ØØ52Ø3                          INC     R3              ;AND R5 FOR DEST ODD BYTE
        Ø1161Ø  Ø1Ø3Ø5                          MOV     R3,R5
        Ø11612  ØØØ261                          SEC                     ;SET CARRY
        Ø11614  Ø12762  125252  Ø1174Ø          MOV     #125252,SDATAB(2)
        Ø11622  112763  177125  Ø1174Ø          MOVB    #177125,SDATAB(3)       ;SOURCE DATA = Ø52652
        Ø1163Ø  Ø16264  Ø1174Ø  Ø11742          MOV     SDATAB(2),DDATAB(4)
        Ø11636  Ø52764  125125  Ø11742          BIS     #125125,DDATAB(4)       ;DEST DATA = 177777
        Ø11644  136263  Ø1174Ø  Ø1174Ø          BITB    SDATAB(2),SDATAB(3)
        Ø11652  ØØ14Ø1                          BEQ     .+4
        Ø11654  1Ø44ØØ                  BITB6:  HLT

        Ø11656  146264  Ø1174Ø  Ø11742          BICB    SDATAB(2),DDATAB(4)
        Ø11664  1Ø34Ø1                          BCS     .+4
        Ø11666  1Ø44ØØ                          HLT                     ;ERROR MOV,BIS,BIT;BIC DO NOT AFFECT 'C'
        Ø1167Ø  126364  Ø1174Ø  Ø11742          CMPB    SDATAB(3),DDATAB(4)
        Ø11676  ØØ14Ø1                          BEQ     .+4
        Ø117ØØ  1Ø44ØØ                          HLT

        Ø117Ø2  146365  Ø1174Ø  Ø11742          BICB    SDATAB(3),DDATAB(5)
        Ø1171Ø  126265  Ø1174Ø  Ø11742          CMPB    SDATAB(2),DDATAB(5)
        Ø11716  ØØ14Ø1                          BEQ     .+4
        Ø1172Ø  1Ø44ØØ                          HLT

        Ø11722  136564  Ø11742  Ø11742          BITB    DDATAB(5),DDATAB(4)
        Ø1173Ø  ØØ14Ø1                          BEQ     .+4
        Ø11732  1Ø44ØØ                          HLT
```

```
        Ø11734  1Ø4ØØØ                          SCOPE

        Ø11736  ØØØ4Ø6                          BR      UB7             ;RESERVE TWO WORDS
        Ø1174Ø  ØØØØØØ                  SDATAB: .WORD   Ø               ;RESERVED FOR SOURCE DATA
        Ø11742  ØØØØØØ                  DDATAB: .WORD   Ø               ;RESERVED FOR DEST DATA

                                        ;CHECK BINARY WORD OPS USING ADDRESS MODE 7
                                        ;R2=ADDRESS OF SOURCE DATA, AND R3= ADDRESS OF DEST DATA
        Ø11744  ØØØØØØ                  SBIN7:  .WORD   Ø               ;CONTAINS ADDRESS OF SOURCE DATA
        Ø11746  ØØØØØØ                  DBIN7:  .WORD   Ø               ;CONTAINS ADDRESS OF DEST DATA
        Ø1175Ø  ØØØØØØ                          .WORD   Ø               ;CONTAINS SOURCE DATA
        Ø11752  ØØØØØØ                          .WORD   Ø               ;CONTAINS DEST DATA

        Ø11754  Ø1Ø7ØØ                  UB7:    MOV     PC,RØ
        Ø11756  Ø24Ø4Ø                          CMP     -(RØ),-(RØ)
        Ø1176Ø  Ø1ØØØ2                          MOV     RØ,R2
        Ø11762  Ø24242                          CMP     -(R2),-(R2)
        Ø11764  Ø1ØØ12                          MOV     RØ,(R2)
        Ø11766  Ø1Ø2Ø3                          MOV     R2,R3
        Ø1177Ø  Ø24Ø43                          CMP     -(RØ),-(R3)
        Ø11772  Ø1ØØ13                          MOV     RØ,(R3)

        Ø11774  ØØØ261                          SEC
        Ø11776  Ø12777  1ØØØØØ  17774Ø          MOV     #1ØØØØØ,@SBIN7  ;SOURCE DATA = 1ØØØØØ
        Ø12ØØ4  Ø17777  177734  177734          MOV     @SBIN7,@DBIN7   ;DEST DATA = 1ØØØØØ
        Ø12Ø12  1Ø3ØØ1                          BCC     MOV7
        Ø12Ø14  1ØØ4Ø1                          BMI     .+4
        Ø12Ø16  1Ø44ØØ                  MOV7:   HLT
        Ø12Ø2Ø  ØØ6377  177722                  ASL     @DBIN7          ;DEST DATA = ØØØØØØ
        Ø12Ø24  1Ø2ØØ1                          BVC     .+4
        Ø12Ø26  ØØ14Ø1                          BEQ     .+4
        Ø12Ø3Ø  1Ø44ØØ                          HLT

        Ø12Ø32  Ø27777  1777Ø6  1777Ø6          CMP     @SBIN7,@DBIN7   ;(R2)=1ØØØØØ,(R3)=ØØØØØØ
        Ø12Ø4Ø  1Ø34Ø2                          BCS     CMP7
        Ø12Ø42  1Ø24Ø1                          BVS     CMP7
        Ø12Ø44  1ØØ4Ø1                          BMI     .+4
        Ø12Ø46  1Ø44ØØ                  CMP7:   HLT

        Ø12Ø5Ø  167777  17767Ø  17767Ø          SUB     @SBIN7,@DBIN7   ;(R2)=1ØØØØØ,(R3)=1ØØØØØ
        Ø12Ø56  1Ø3ØØ3                          BCC     SUB7
        Ø12Ø6Ø  1Ø2ØØ2                          BVC     SUB7
        Ø12Ø62  ØØ14Ø1                          BEQ     SUB7
        Ø12Ø64  1ØØ4Ø1                          BMI     .+4
        Ø12Ø66  1Ø44ØØ                  SUB7:   HLT

        Ø12Ø7Ø  ØØ6277  17765Ø                  ASR     @SBIN7          ;(R2)=14ØØØØ
        Ø12Ø74  Ø67777  177644  177644          ADD     @SBIN7,@DBIN7   ;(R2)=14ØØØØ,(R3)=Ø4ØØØØ
        Ø121Ø2  1Ø3ØØ3                          BCC     ADD7
        Ø121Ø4  1Ø2ØØ2                          BVC     ADD7
        Ø121Ø6  ØØ14Ø1                          BEQ     ADD7
        Ø1211Ø  1ØØØØ1                          BPL     .+4
        Ø12112  1Ø44ØØ                  ADD7:   HLT
        Ø12114  Ø47777  177624  177624          BIC     @SBIN7,@DBIN7   ;(R2)=14ØØØØ,(R3)=ØØØØØØ
        Ø12122  ØØ14Ø1                          BEQ     .+4
        Ø12124  1Ø44ØØ                          HLT

        Ø12126  Ø57777  177612  177612          BIS     @SBIN7,@DBIN7   ;(R2)=14ØØØØ,(R3)=14ØØØØ
        Ø12134  1ØØ4Ø1                          BMI     .+4
        Ø12136  1Ø44ØØ                          HLT

        Ø1214Ø  Ø27777  1776ØØ  1776ØØ          CMP     @SBIN7,@DBIN7
        Ø12146  ØØ14Ø1                          BEQ     .+4
        Ø1215Ø  1Ø44ØØ                          HLT
        Ø12152  1Ø4ØØØ                          SCOPE

                                        ;SOME MISCELLANEOUS OPERATION INVOLVING THE PC
                                        ;NOTE: NONE OF THESE OPERATIONS SHOULD AFFECT THE PC
        Ø12154  ØØ5ØØØ                          CLR     RØ
        Ø12156  ØØ5Ø67  ØØØØ72                  CLR     1$
        Ø12162  Ø1Ø7Ø7                          MOV     PC,PC
        Ø12164  12Ø7Ø7                          CMPB    PC,PC
        Ø12166  Ø3Ø7Ø7                          BIT     PC,PC
        Ø1217Ø  Ø6ØØØ7                          ADD     RØ,PC
        Ø12172  1Ø57Ø7                          TSTB    PC
        Ø12174  ØØ55Ø7                          ADC     PC
        Ø12176  Ø21ØØ7                          CMP     (RØ),PC
        Ø122ØØ  131ØØ7                          BITB    (RØ),PC
        Ø122Ø2  Ø627Ø7  ØØØØØØ                  ADD     #Ø,PC
        Ø122Ø6  Ø237Ø7  ØØ1ØØ4                  CMP     @#FACTOR,PC
        Ø12212  1337Ø7  ØØ1ØØ4                  BITB    @#FACTOR,PC
        Ø12216  ØØØ24Ø                          NOP
                                        ;THE NEXT TWO INSTRUCTION CAUSE THE PROGRAM TO JUMP TO THE UNRELOCATED
                                        ;CODE AND TO RETURN ON THE FOLLOWING INST (IF THE CODE IS RELOCATED)
        Ø1222Ø  1637Ø7  ØØ1ØØ4                  SUB     @#FACTOR,PC     ;JUMPS TO UNRELOCATED CODE
        Ø12224  Ø637Ø7  ØØ1ØØ4                  ADD     @#FACTOR,PC     ;RETURNS
        Ø1223Ø  ØØØ24Ø                          NOP
        Ø12232  Ø246Ø7                          CMP     -(SP),PC
        Ø12234  1326Ø7                          BITB    (SP)+,PC
        Ø12236  Ø267Ø7  ØØØØ12                  CMP     1$,PC
        Ø12242  1667Ø7  ØØØØØ6                  SUB     1$,PC
        Ø12246  Ø467Ø7  ØØØØØ2                  BIC     1$,PC
        Ø12252  ØØØ4Ø1                          BR      .+4             ;BRANCH OVER 1$
        Ø12254  ØØØØØØ                  1$:     Ø
        Ø12256  1Ø4ØØØ                          SCOPE

        Ø1226Ø  Ø1Ø7Ø2                          MOV     PC,R2
        Ø12262  Ø627Ø2  ØØØØ12                  ADD     #12,R2
        Ø12266  Ø127Ø7  ØØ1132                  MOV     #RELOC,PC       ;GO RELOCATE PROGRAM CODE
        Ø12272  ØØØ24Ø                          NOP                     ;PROGRAM RETURNS HERE+2
                                        ;1111111111111 LAST ADDRESS OF CODE TO BE RELOCATED 1111111111


                                        ;2222222222222 FIRST ADDRESS TO BE RELOCATED 222222222
        Ø12274  Ø1Ø7ØØ                  REL2:   MOV     PC,RØ           ;GET PC
        Ø12276  ØØ574Ø                          TST     -(RØ)           ;RØ CONTAINS THE ADDRESS OF REL2
        Ø123ØØ  Ø1ØØ37  ØØ1Ø1Ø                  MOV     RØ,@#FRSTAD     ;SAVE
```

```
        Ø123Ø4  Ø1Ø7ØØ                          MOV     PC,RØ           ;GET CURRENT PC
        Ø123Ø6  1627ØØ  Ø123Ø6                  SUB     #.,RØ           ;SUBTRACT RELOCATION FACTOR
        Ø12312  Ø1ØØ37  ØØ1ØØ4                  MOV     RØ,@#FACTOR     ;SAVE RELOCATION FACTOR
        Ø12316  Ø1Ø7Ø1                          MOV     PC,R1           ;SET NEW SCOPE PTR
                                        ;CHECK BINARY BYTE OPS USING ADDRESS MODE 7
        Ø1232Ø  ØØØ4Ø6                          BR      BINB7           ;RESERVE SPACE FOR ADDRESSES & DATA
        Ø12322  ØØØØØØ                  SBINB7: .WORD   Ø               ;CONTAINS ADDRESS OF SOURCE EVEN BYTE
        Ø12324  ØØØØØØ                          .WORD   Ø               ;CONTAINS ADDRESS OF SOURCE ODD BYTE
        Ø12326  ØØØØØØ                          .WORD   Ø               ;CONTAINS ADDRESS OF DEST EVEN BYTE
        Ø1233Ø  ØØØØØØ                          .WORD   Ø               ;CONTAINS ADDRESS OF DEST ODD BYTE
        Ø12332  ØØØØØØ                  DBINB7: .WORD   Ø               ;CONTAINS SOURCE DATA
        Ø12334  ØØØØØØ                          .WORD   Ø               ;CONTAINS DEST DATA

        Ø12336  Ø1Ø7ØØ                  BINB7:  MOV     PC,RØ
        Ø1234Ø  Ø24Ø4Ø                          CMP     -(RØ),-(RØ)     ;RØ = ADDRESS OF DEST DATA
        Ø12342  Ø1ØØ6Ø  177772                  MOV     RØ,-6(RØ)       ;LOAD ADDRES OF DEST EVEN BYTE DATA
        Ø12346  Ø1ØØ6Ø  177774                  MOV     RØ,-4(RØ)
        Ø12352  ØØ526Ø  177774                  INC     -4(RØ)          ;LOAD ADDRESS OF DEST ODD BYTE DATA
        012356  005740                          TST     -(R0)           ;R0=ADDRESS OF SOURCE DATA
        012360  010060  177770                  MOV     R0,-10(R0)      ;LOAD ADDRESS OF SOURCE EVEN BYTE DATA
        012364  010060  177772                  MOV     R0,-6(R0)
        012370  005260  177772                  INC     -6(R0)          ;LOAD ADDRESS OF SOURCE ODD BYTE DATA

        012374  005002                          CLR     R2              ;SET INDEX REGISTERS
        012376  012703  000002                  MOV     #2,R3           ;@SBINB7(2);@SBINB7(3) REFERENCE EVEN &
        012402  012704  177774                  MOV     #-4,R4          ;ODD BYTE SOURCE DATA; @DBINB7(4);@DBINB7(5)
        012406  012705  177776                  MOV     #-2,R5          ;REFERENCE DEST EVEN& ODD BYTE DATA


        012412  005020                          CLR     (R0)+           ;PRESET SOURCE DATA
        012414  005010                          CLR     (R0)            ;PRESET DEST DATA
        012416  013746  001004                  MOV     @#FACTOR,-(SP)  ;GET RELOCATION FACTOR
        012422  061602                          ADD     (SP),R2         ;AND ADD TO INDEX VALUES
        012424  061603                          ADD     (SP),R3
        012426  061604                          ADD     (SP),R4
        012430  062605                          ADD     (SP)+,R5

        012432  112773  177777  012322          MOVB    #-1,@SBINB7(3)  ;SRC DATA = 177400
        012440  132772  000377  012322          BITB    #377,@SBINB7(2) ;CHECK THAT EVEN BYTE WAS NOT AFFECTED
        012446  001401                          BEQ     .+4             ;BY MOVB INSTRUCTION
        012450  104400                          HLT

        012452  157374  012322  012332          BISB    @SBINB7(3),@DBINB7(4)
        012460  105274  012332                  INCB    @DBINB7(4)      ;CHECK THAT BIS SET ALL BITS
        012464  001401                          BEQ     .+4
        012466  104400                          HLT

        012470  105375  012332                  DECB    @DBINB7(5)      ;DEST DATA = 177400
        012474  005274  012332                  INC     @DBINB7(4)      ;DEST DATA = 177401
        012500  127375  012322  012332          CMPB    @SBINB7(3),@DBINB7(5)
        012506  001401                          BEQ     .+4
        012510  104400                          HLT

        012512  147375  012322  012332          BICB    @SBINB7(3),@DBINB7(5)
        012520  001401                          BEQ     .+4
        012522  104400                          HLT

        012524  105073  012322                  CLRB    @SBINB7(3)      ;SRC DATA = 000000
                                        ;THIS ROUTINE SETS ALL BITS IN THE SOURCE ODD BYTE BY BISING A BIT FROM
                                        ;THE DEST EVEN BYTE INTO THE SOURCE ODD BYTE
        012530  157473  012332  012322  BIS7:   BISB    @DBINB7(4),@SBINB7(3)
        012536  106174  012332                  ROLB    @DBINB7(4)
        012542  103372                          BCC     BIS7
        012544  022772  177400  012322          CMP     #177400,@SBINB7(2)      ;CHECK RESULT
        012552  001401                          BEQ     .+4
        012554  104400                          HLT

        012556  000372  012322                  SWAB    @SBINB7(2)      ;SRC DATA = 000377
        012562  112775  000200  012332          MOVB    #200,@DBINB7(5) ;DEST DATA = 100000

        012570  147572  012332  012322  BIC7:   BICB    @DBINB7(5),@SBINB7(2)
        012576  106075  012332                  RORB    @DBINB7(5)
```

```
        012602  103372                          BCC     BIC7
        012604  005772  012322                  TST     @SBINB7(2)
        012610  001401                          BEQ     .+4
        012612  104400                          HLT
        012614  104000                          SCOPE

        012616  012702  000001          OAERR:  MOV     #1,R2           ;LOAD R2 WITH ODD #
        012622  010703                          MOV     PC,R3
        012624  000401                          BR      .+4             ;RESERVE SPACE FOR A WORD
        012626  000000                          .WORD   0               ;WILL CONTAIN AN ODD ADDRESS
        012630  005723                          TST     (R3)+           ;STEP R3 TO POINT TO WORD ABOVE
        012632  010313                          MOV     R3,(R3)
        012634  005213                          INC     (R3)            ;AND MAKE ODD
        012636  012737  012764  000004          MOV     #1$,@#ERRVEC    ;SET ODD ADDRESS & RESERVED INSTRUCTION
        012644  063737  001004  000004          ADD     @#FACTOR,@#ERRVEC
        012652  013737  000004  000010          MOV     @#ERRVEC,@#RESVEC       ;TO TRAP TO 1$ BELOW

        012660  000277                          SCC                     ;SET ALL CC'S
        012662  160212                          SUB     R2,(R2)
        012664  104400                          HLT
Z       012666  060222                          ADD     R2,(R2)+
        012670  104400                          HLT
        012672  006342                          ASL     -(R2)
        012674  104400                          HLT
        012676  106512                          MFPD    (R2)            ;NOTE: MAY BE RESERVED
        012700  104400                          HLT
        012702  170412                          CLRF    (R2)
        012704  104400                          HLT
        012706  042202                          BIC     (R2)+,R2
        012710  104400                          HLT
        012712  164202                          SUB     -(R2),R2
        012714  104400                          HLT
        012716  155202                          BISB    @-(R2),R2
        012720  104400                          HLT
        012722  105532                          ADCB    @(R2)+
        012724  104400                          HLT
        012726  163302                          SUB     @(R3)+,R2
        012730  104400                          HLT
        012732  005733                          TST     @(R3)+
        012734  104400                          HLT
        012736  106533                          MFPD    @(R3)+
        012740  104400                          HLT
        012742  170453                          CLRD    @-(R3)
        012744  104400                          HLT
        012746  137702  177775                  BITB    @.+1,R2
        012752  104400                          HLT
        012754  105477  177773                  NEGB    @.-1
        012760  104400                          HLT
        012762  000406                          BR      2$

        012764  062716  000002          1$:     ADD     #2,(SP)         ;ADJUST RETURN PC
        012770  052766  000017  000002          BIS     #17,2(SP)       ;SET CONDITION CODES ON RETURN
        012776  000002                          RTI
        Ø13ØØØ  Ø127Ø6  ØØØ5ØØ                  2$:     MOV     #STKPTR,SP      ;RESET STACK PTR
        Ø13ØØ4  Ø12737  ØØØØØ6  ØØØØØ4                  MOV     #ERRVEC+2,@#ERRVEC
        Ø13Ø12  Ø12737  ØØØØ12  ØØØØ1Ø                  MOV     #RESVEC+2,@#RESVEC
        Ø13Ø2Ø  1Ø4ØØØ                                  SCOPE

                                                ;CHECK JMP INSTRUCTIONS

        Ø13Ø22  Ø1Ø7ØØ                                  MOV     PC,RØ
        Ø13Ø24  Ø627ØØ  ØØØØ12                          ADD     #12,RØ          ;SET ADDRESS FOR JMP INST
        Ø13Ø3Ø  ØØØ277                                  SCC                     ;SET CC'S
        Ø13Ø32  ØØØ11Ø                                  JMP     (RØ)
        Ø13Ø34  ØØØ4Ø2                                  BR      .+6
        Ø13Ø36  ØØØ25Ø                                  CLN                     ;JMP INST JUMPS HERE
        Ø13Ø4Ø  ØØØ775                                  BR      .-4

        Ø13Ø42  1Ø3ØØ3                                  BCC     JMP1
        Ø13Ø44  1Ø2ØØ2                                  BVC     JMP1
        Ø13Ø46  ØØ1ØØ1                                  BNE     JMP1
        Ø13Ø5Ø  1ØØØØ1                                  BPL     .+4
        Ø13Ø52  1Ø44ØØ                          JMP1:   HLT                     ;ERROR! INCORRECT CC'S AFTER JMP

        Ø13Ø54  ØØ5ØØ2                                  CLR     R2              ;SET INDICATOR
        Ø13Ø56  Ø1Ø7Ø3                                  MOV     PC,R3
        Ø13Ø6Ø  ØØØ4Ø1                                  BR      .+4             ;RESERVE WORD FOR JMP ADDRESS
        Ø13Ø62  ØØØØØØ                                  .WORD   Ø               ;CONTAINS ADDRESS FOR JMP INST
        Ø13Ø64  ØØ5723                                  TST     (R3)+
        Ø13Ø66  Ø1Ø313                                  MOV     R3,(R3)
        Ø13Ø7Ø  Ø1Ø3ØØ                                  MOV     R3,RØ
        Ø13Ø72  Ø62713  ØØØØ22                          ADD     #22,(R3)        ;(R3) IS JMP ADDRESS
        Ø13Ø76  Ø1Ø3ØØ                                  MOV     R3,RØ
        Ø131ØØ  ØØØ133                                  JMP     @(R3)+          ;JUMP TO ADDRESS CONTAINED IN R3
        Ø131Ø2  ØØØ4Ø2                                  BR      .+6
        Ø131Ø4  ØØ51Ø2                                  COM     R2              ;COMPLEMENT INDICATOR
        Ø131Ø6  ØØØ775                                  BR      .-4
        Ø1311Ø  ØØ52Ø2                                  INC     R2              ;CHECK INDICATOR
        Ø13112  ØØ1ØØ3                                  BNE     JMP3
        Ø13114  ØØ572Ø                                  TST     (RØ)+
        Ø13116  Ø2ØØØ3                                  CMP     RØ,R3           ;CHECK AUTO-INC R3
        Ø1312Ø  ØØ14Ø1                                  BEQ     .+4
        Ø13122  1Ø44ØØ                          JMP3:   HLT

        Ø13124  ØØ5ØØ2                                  CLR     R2              ;SET INDICATOR
        Ø13126  Ø1Ø7Ø4                                  MOV     PC,R4           ;SET UP JMP REGISTER
        Ø1313Ø  Ø1Ø4ØØ                                  MOV     R4,RØ           ;SET UP CHECK REGISTER
        Ø13132  ØØØ4Ø2                                  BR      1$
        Ø13134  ØØ51Ø2                                  COM     R2              ;COMPLEMENT INDICATOR
        Ø13136  ØØØ4Ø3                                  BR      2$
        Ø1314Ø  Ø22424                          1$:     CMP     (R4)+,(R4)+
        Ø13142  ØØ5724                                  TST     (R4)+           ;R4=JMP ADDRESS
        Ø13144  ØØØ144                                  JMP     -(R4)           ;USE R4 AS ADDRESS
        Ø13146  ØØ52Ø2                          2$:     INC     R2              ;CHECK INDICATOR
        Ø1315Ø  ØØ1ØØ3                                  BNE     JMP4
        Ø13152  Ø22Ø2Ø                                  CMP     (RØ)+,(RØ)+
        Ø13154  Ø2ØØØ4                                  CMP     RØ,R4           ;CHECK AUTO-DEC R4
```

```
        Ø13156  ØØ14Ø1                                  BEQ     .+4
        Ø1316Ø  1Ø44ØØ                          JMP4:   HLT

        Ø13162  Ø1Ø7Ø3                                  MOV     PC,R3
        Ø13164  ØØØ4Ø1                                  BR      .+4             ;RESERVE WORD FOR JMP ADDRESS
        Ø13166  ØØØØØØ                          1$:     .WORD   Ø               ;CONTAINS JUMP ADDRESS
        Ø1317Ø  ØØ5723                                  TST     (R3)+
        Ø13172  Ø1Ø313                                  MOV     R3,(R3)
        Ø13174  Ø62723  ØØØØ16                          ADD     #16,(R3)+
        Ø132ØØ  Ø1Ø3ØØ                                  MOV     R3,RØ           ;LOAD CHECK REGISTER
        Ø132Ø2  ØØØ4Ø2                                  BR      3$
        Ø132Ø4  ØØ51Ø2                          2$:     COM     R2
        Ø132Ø6  ØØØ4Ø1                                  BR      4$
        Ø1321Ø  ØØØ153                          3$:     JMP     @-(R3)          ;JUMP TO 2$ VIA 1$ ABOVE
        Ø13212  ØØ52Ø2                          4$:     INC     R2              ;CHECK INDICATOR
        Ø13214  ØØ1ØØ3                                  BNE     JMP5
        Ø13216  ØØ574Ø                                  TST     -(RØ)
        Ø1322Ø  Ø2ØØØ3                                  CMP     RØ,R3           ;CHECK AUTO-DEC R3
        Ø13222  ØØ14Ø1                                  BEQ     .+4
        Ø13224  1Ø44ØØ                          JMP5:   HLT

        Ø13226  ØØØ4Ø2                                  BR      2$
        Ø1323Ø  ØØ51Ø2                          1$:     COM     R2              ;COMPLEMENT INDICATOR
        Ø13232  ØØØ4Ø2                                  BR      3$
        Ø13234  ØØØ167  17777Ø                  2$:     JMP     1$
        Ø1324Ø  ØØ52Ø2                          3$:     INC     R2
        Ø13242  ØØ14Ø1                                  BEQ     .+4
        Ø13244  1Ø44ØØ                          JMP6:   HLT

        Ø13246  Ø12767  Ø13264  ØØØØ2Ø                  MOV     #1$,7$          ;SET UP JMP ADDRESS
        Ø13254  Ø63767  ØØ1ØØ4  ØØØØ12                  ADD     @#FACTOR,7$     ;ADD RELOCATION FACTOR
        Ø13262  ØØØ4Ø2                                  BR      2$              ;GO TO JMP  @7$ INST
        Ø13264  ØØ51Ø2                          1$:     COM     R2              ;COMPLEMENT INDICATOR
        Ø13266  ØØØ4Ø3                                  BR      3$              ;GO TO CHECK ROUTINE
        Ø1327Ø  ØØØ177  ØØØØØØ                  2$:     JMP     @7$             ;JMP TO 1$ ABOVE VIA 7$
        Ø13274  ØØØØØØ                          7$:     .WORD   Ø               ;CONTAINS JMP ADDRESS
        Ø13276  ØØ52Ø2                          3$:     INC     R2              ;CHECK INDICATOR
        Ø133ØØ  ØØ14Ø1                                  BEQ     .+4
        Ø133Ø2  1Ø44ØØ                          JMP7:   HLT
        Ø133Ø4  1Ø4ØØØ                                  SCOPE

                                                ;CHECK JSR INSTRUCTIONS
        Ø133Ø6  Ø137Ø5  ØØ1ØØ4                  JSRTST: MOV     @#FACTOR,R5     ;GET RELOCATION FACTOR
        Ø13312  Ø127Ø2  Ø13344                          MOV     #3$,R2          ;FORM DEST ADRS
        Ø13316  Ø6Ø5Ø2                                  ADD     R5,R2           ;ADD RELOCATION FACTOR
        Ø1332Ø  ØØØ277                                  SCC                     ;PRESET CC'S
        Ø13322  ØØØ242                                  CLV
        Ø13324  ØØ4512                                  JSR     R5,(R2)         ;GO TO 3$ VIA R2
        Ø13326  ØØ57Ø2                          1$:     TST     R2              ;CHECK INDICATOR
        Ø1333Ø  ØØ1Ø17                                  BNE     JSR1            ;R2 SHOULD=Ø
        Ø13332  Ø237Ø5  ØØ1ØØ4                          CMP     @#FACTOR,R5     ;CHECK THAT RTS  R5 RESTORED R5
        Ø13336  ØØ1Ø14                                  BNE     JSR1
        Ø1334Ø  ØØØ414                                  BR      JSR1A           ;EXIT TO SCOPE
        Ø13342  ØØØ2Ø5                          2$:     RTS     R5              ;RETURN FROM SUBROUTINE
        013344  103011                    3$:     BCC     JSR1            ;CHECK THAT JSR DID NOT
        013346  102410                            BVS     JSR1
        013350  001007                            BNE     JSR1            ;AFFECT CC'S
        013352  100006                            BPL     JSR1
        013354  005002                            CLR     R2              ;CLEAR INDICATOR
        013356  012704  013326                    MOV     #1$,R4          ;GET UNRELOCATED RETURN ADDRESS
        013362  061604                            ADD     (SP),R4         ;ADD RELOCATION FACTOR (OLD R5)
        013364  020405                            CMP     R4,R5           ;CHECK THAT OLD R5 WAS PLACED ON THE
        013366  001765                            BEQ     2$              ;STACK,& THAT NEW R5 CONTAINS RETURN PC
        013370  104400                    JSR1:   HLT                     ;ERROR! ABOVE

        013372  013704  001004            JSR1A:  MOV     @#FACTOR,R4     ;GET RELOCATION FACTOR
        013376  005000                            CLR     R0              ;SET INDICATOR
        013400  012705  013420                    MOV     #1$,R5
        013404  060405                            ADD     R4,R5           ;SET UP JSR DEFERRED ADRS
        013406  010502                            MOV     R5,R2
        013410  012715  013436                    MOV     #5$,(R5)
        013414  060415                            ADD     R4,(R5)         ;(R5)=DEST ADRS
        013416  000401                            BR      2$              ;RESERVE WORD FOR ADDRESS
        013420  000000                    1$:     .WORD   0               ;CONTAINS DEST ADRS FOR JSR
        013422  004435                    2$:     JSR     R4,@(R5)+       ;JSR TO 5$ VIA 1$ ABOVE
        013424  005200                    3$:     INC     R0              ;CHECK INDICATOR
        013426  001013                            BNE     JSR3
        013430  000413                            BR      JSR3A
        013432  005100                    4$:     COM     R0              ;COMPLEMENT INDICATOR
        013434  000204                            RTS     4               ;RETURN FROM SUBROUTINE
        013436  012703  013424            5$:     MOV     #3$,R3          ;GET UNRELOCATED RETURN ADDRESS
        013442  061603                            ADD     (SP),R3         ;ADD RELOCATION FACTOR (OLD R4)
        013444  020403                            CMP     R4,R3
        013446  001003                            BNE     JSR3
        013450  005722                            TST     (R2)+
        013452  020205                            CMP     R2,R5           ;CHECK AUTO-INC R5
        013454  001766                            BEQ     4$              ;GO TO RTS
        013456  104400                    JSR3:   HLT                     ;ERROR ABOVE

        013460  013704  001004            JSR3A:  MOV     @#FACTOR,R4
        013464  010405                            MOV     R4,R5
        013466  010703                            MOV     PC,R3
        013470  000401                            BR      2$
        013472  000405                    1$:     BR      4$
        013474  022323                    2$:     CMP     (R3)+,(R3)+
        013476  000277                            SCC
        013500  004443                            JSR     R4,-(R3)                ;GO TO 2$
        013502  104400                    3$:     HLT
        013504  000414                            BR      JSR4A
        013506  103012                    4$:     BCC     JSR4
        013510  102011                            BVC     JSR4
        013512  001010                            BNE     JSR4
        013514  100007                            BPL     JSR4
        013516  012702  013502                    MOV     #3$,R2          ;GET UNRELOCATED RETURN ADDRESS
        013522  061602                            ADD     (SP),R2         ;ADD RELOCATION FACTOR (OLD R4)
        013524  020204                            CMP     R2,R4           ;CHECK THAT CALCULATED RETURN
        013526  001002                            BNE     JSR4            ;PC = NEW R4
        013530  005724                            TST     (R4)+
```

```
        013532  000204                            RTS     R4
        013534  104400                    JSR4:   HLT


        013536  000401                    JSR4A:  BR      2$
        013540  000405                    1$:     BR      3$
        013542  010700                    2$:     MOV     PC,R0
        013544  004767  177770                    JSR     PC,1$
        013550  100407                            BMI     JSR6A
        013552  104400                            HLT
        013554  022020                    3$:     CMP     (R0)+,(R0)+
        013556  020016                            CMP     R0,(SP)         ;CHECK THAT RETURN ADDRESS IS ON THE
        013560  001401                            BEQ     .+4             ;STACK
        013562  104400                            HLT
        013564  000270                            SEN                     ;SET N
        013566  000207                            RTS     PC
        013570  104000                    JSR6A:  SCOPE

                                          ;CHECK IOT TRAP (AND ROLB/ASLB)
        013572  012737  013624  000020            MOV     #IOT1,@#IOTVEC
        013600  063737  001004  000020            ADD     @#FACTOR,@#IOTVEC       ;ADD RELOCATION FACTOR
        013606  000261                            SEC                             ;SET CARRY
        013610  013737  177776  000022            MOV     @#PSW,@#IOTVEC+2        ;RETAIN CURRENT PSW ON TRAP
        013616  005000                            CLR     R0                      ;PRESET R0
        013620  000004                            IOT
        013622  000403                            BR      IOT1A
        013624  106100                    IOT1:   ROLB    R0                      ;ROTATE R0
        013626  102376                            BVC     .-2             ;UNTIL V SETS (R0=200)
        013630  000002                            RTI
        013632  106300                    IOT1A:  ASLB    R0              ;SHIFT SHOULD SET CARRY
        013634  103004                            BCC     IOT1B
        013636  102003                            BVC     IOT1B
        013640  001002                            BNE     IOT1B
        013642  005700                            TST     R0              ;R0 SHOULD =0
        013644  001401                            BEQ     .+4
        013646  104400                    IOT1B:  HLT                     ;ERROR! ROL/ASL FAILED TO SET CC'S PROPERLY
        013650  012737  000022  000020            MOV     #IOTVEC+2,@#IOTVEC      ;RESTORE IOT TRAP
        013656  005037  000022                    CLR     @#IOTVEC+2              ;VECTOR
        013662  104000                            SCOPE

                                          ;CHECK EMT TRAP SEQUENCE
        013664  013746  000030                    MOV     @#EMTVEC,-(SP)  ;SAVE SCOPE PTR
        013670  012737  013724  000030            MOV     #EMT1,@#EMTVEC  ;SET EMT TRAP VECTOR
        013676  063737  001004  000030            ADD     @#FACTOR,@#EMTVEC       ;ADD RELOCATION FACTOR
        013704  000262                            SEV                     ;SET V
        013706  013737  177776  000032            MOV     @#PSW,@#EMTVEC+2        ;RETAIN CURRENT PSW ON TRAP
        013714  000265                            +SEZ!SEC
        013716  104000                            EMT                     ;TRAP TO EMT1
        013720  001433                            BEQ     EMT1C           ;GO TO EMT1C
        013722  104400                            HLT                     ;ERROR! INCORRECT CC'S WERE SET ON RETURN
        013724  102027                    EMT1:   BVC     EMT1B           ;'V' SHOULD'VE SET ON EMT TRAP
        013726  105100                            COMB    R0              ;R0=000377,CC'S=1001
        013730  105500                            ADCB    R0              ;R0=000000,CC'S=0101
        013732  106000                            RORB    R0              ;R0=000200,CC'S=1010
        Ø13734  1Ø2Ø23                          BVC     EMT1B
        Ø13736  1ØØØ22                          BPL     EMT1B
        Ø1374Ø  ØØØ257                          CCC
        Ø13742  1054ØØ                          NEGB    RØ              ;RØ=ØØØ2ØØ,CC'S=1Ø1Ø
        Ø13744  1Ø2Ø17                          BVC     EMT1B
        Ø13746  1ØØØ16                          BPL     EMT1B
        Ø1375Ø  ØØØ242                          CLV                     ;CLEAR 'V'
        Ø13752  ØØØ261                          SEC                     ;AND SET 'C'
        Ø13754  1053ØØ                          DECB    RØ              ;RØ=ØØØ177,CC'S=ØØ11
        Ø13756  1Ø2Ø12                          BVC     EMT1B
        Ø1376Ø  1ØØ411                          BMI     EMT1B
        Ø13762  ØØØ242                          CLV                     ;CLEAR 'V'
        Ø13764  1052ØØ                          INCB    RØ              ;RØ=ØØØ2ØØ,CC'S=1Ø11
        Ø13766  1Ø3ØØ6                          BCC     EMT1B
        Ø1377Ø  1Ø2ØØ5                          BVC     EMT1B
        Ø13772  1ØØØØ4                          BPL     EMT1B
        Ø13774  ØØØ242                          CLV                     ;CLEAR 'V'
        Ø13776  1062ØØ                          ASRB    RØ              ;SHIFT RØ UNTIL 'V' CLEARS
        Ø14ØØØ  1Ø2776                          BVS     .-2
        Ø14ØØ2  ØØØ4Ø1                          BR      .+4
        Ø14ØØ4  1044ØØ                  EMT1B:  HLT                     ;ERROR!
        Ø14ØØ6  ØØØØØ2                          RTI                     ;EXIT WITH RØ=ØØØ377
        Ø14Ø1Ø  1055ØØ                  EMT1C:  ADCB    RØ              ;RØ=ØØØØØØ
        Ø14Ø12  1Ø3ØØ3                          BCC     EMT1D
        Ø14Ø14  ØØ1ØØ2                          BNE     EMT1D
        Ø14Ø16  ØØ57ØØ                          TST     RØ
        Ø14Ø2Ø  ØØ14Ø1                          BEQ     .+4
        Ø14Ø22  1044ØØ                  EMT1D:  HLT
        Ø14Ø24  Ø12637  ØØØØ3Ø                  MOV     (SP)+,@#EMTVEC  ;RESTORE SCOPE PTR
        Ø14Ø3Ø  ØØ5Ø37  ØØØØ32                  CLR     @#EMTVEC+2
        Ø14Ø34  1Ø4ØØØ                          SCOPE

                                        ;CHECK TRAP INSTRUCTION TRAP SEQUENCE
                ØØØØØ4                          HLT=IOT                 ;REDEFINE HLT
        Ø14Ø36  Ø13737  ØØØØ34  ØØØØ2Ø          MOV     @#TRAPVEC,@#IOTVEC      ;SET IOT (HLT) TRAP VECTOR
        Ø14Ø44  Ø12737  Ø14112  ØØØØ34          MOV     #TRAP1,@#TRAPVEC        ;SET TRAP VECTOR
        Ø14Ø52  Ø63737  ØØ1ØØ4  ØØØØ34          ADD     @#FACTOR,@#TRAPVEC      ;ADD RELOCATION FACTOR
        Ø14Ø6Ø  ØØØ27Ø                          SEN                     ;SET N
        Ø14Ø62  Ø13737  177776  ØØØØ36          MOV     @#PSW,@#TRAPVEC+2       ;RETAIN CURRENT PSW ON TRAP
        Ø14Ø7Ø  ØØØ261                          SEC                             ;SET CARRY
        Ø14Ø72  11Ø7ØØ                          MOVB    PC,RØ
        Ø14Ø74  ØØØ264                          SEZ                     ;SET Z BIT
        Ø14Ø76  1044ØØ                          TRAP                    ;TRAP TO TRAP1
        Ø141ØØ  1Ø34Ø1                  1$:     BCS     .+4
        Ø141Ø2  ØØØØØ4                          HLT
        Ø141Ø4  ØØ14Ø1                          BEQ     .+4
        Ø141Ø6  ØØØØØ4                          HLT
        Ø1411Ø  ØØØ412                          BR      TRAP1C
        Ø14112  1ØØ4Ø1                  TRAP1:  BMI     .+4             ;N BIT GOT SET ON TRAP
        Ø14114  ØØØØØ4                          HLT
        Ø14116  Ø627ØØ  ØØØØØ4                  ADD     #4,RØ
        Ø14122  12ØØ16                          CMPB    RØ,(SP)         ;CHECK LOW BYTE OF RETURN PC ON
        Ø14124  ØØ14Ø1                          BEQ     .+4             ;STACK
        Ø14126  ØØØØØ4                          HLT
```

```
        Ø1413Ø  124646                          CMPB    -(SP),-(SP)
        Ø14132  Ø32626                          BIT     (SP)+,(SP)+
        Ø14134  ØØØØØ2                          RTI                     ;RETURN TO INST FOLLOWING TRAP (1$)

        Ø14136  Ø13737  ØØØØ2Ø  ØØØØ34  TRAP1C: MOV     @#IOTVEC,@#TRAPVEC      ;RESTORE TRAP (HLT) TRAP VECTOR
        Ø14144  Ø12737  ØØØ2ØØ  ØØØØ36          MOV     #PRTY4,@#TRAPVEC+2
        Ø14152  Ø12737  ØØØØ22  ØØØØ2Ø          MOV     #IOTVEC+2,@#IOTVEC
        Ø1416Ø  ØØ5Ø37  ØØØØ22                  CLR     @#IOTVEC+2
        Ø14164  1Ø4ØØØ                          SCOPE
                1044ØØ                          HLT=TRAP                ;RESTORE HLT TO A TRAP INST

        Ø14166  Ø1Ø7Ø2                          MOV     PC,R2
        Ø1417Ø  Ø627Ø2  ØØØØ12                  ADD     #12,R2
        Ø14174  Ø127Ø7  ØØ1132                  MOV     #RELOC,PC       ;GO RELOCATE PROGRAM CODE
        Ø142ØØ  ØØØ24Ø                          NOP                     ;PROGRAM RETURNS HERE+2
                                        ;2222222222222 LAST ADDRESS OF CODE TO BE RELOCATED 2222222222

        Ø142Ø2  Ø1Ø7Ø1                          MOV     PC,R1           ;SET SCOPE PTR

                                        ;THE BELOW ROUTINE ASCERTAINS WHICH CP & CP OPTIONS THE PROGRAM IS RUN-
                                        ;NING ON AND SETS AN INDICATOR IN OPT.CP ACCORDINGLY.
        Ø142Ø4  ØØ5767  16457Ø          CPCHK:  TST     ICNT            ;CHECK IF PASS Ø
        Ø1421Ø  ØØ1Ø36                          BNE     REL3            ;DO NOT EXECUTE ROUTINE IF NOT PASS Ø
        Ø14212  Ø12737  ØØØØØ2  ØØØØØ6          MOV     #RTI,@#ERRVEC+2 ;SET UP ERROR TRAP TO RETURN
        Ø1422Ø  Ø127ØØ  ØØØØØ3                  MOV     #3,RØ
        Ø14224  ØØØ261                          SEC
        Ø14226  ØØ5737  177772                  TST     @#PIRQ          ;RØ=3 IF 11/45
        Ø14232  ØØ56ØØ                          SBC     RØ              ;RØ=2 IF 11/4Ø
        Ø14234  ØØØ261                          SEC
        Ø14236  1Ø5737  177777                  TSTB    @#PSW+1         ;RØ=1 IF 11/2Ø
        Ø14242  ØØ56ØØ                          SBC     RØ
        Ø14244  ØØ5Ø37  1777ØØ                  CLR     @#1777ØØ        ;RØ=Ø IF 11/Ø5
        Ø1425Ø  ØØ63ØØ                          ASL     RØ              ;SHIFT INDICATOR
        Ø14252  Ø1ØØ27                          MOV     RØ,(PC)+        ;SET CP INDICATOR
        Ø14254  ØØØØØØ                  OPT.CP: .WORD   Ø               ;CONTAINS OPTION & CP INDICATORS
                                                ;EVEN BYTE:     Ø=11/Ø5,2=11/2Ø,4=11/4Ø,6=11/45
                                                ;ODD BYTE:      2ØØ=MEM MGMT,1ØØ=EIS, 4Ø=11/45 FLOATING POINT
        Ø14256  ØØ5Ø37  ØØØØØ6          3$:     CLR     @#ERRVEC+2      ;RESTORE ERROR TRAP TO HALT ON TRAP
        Ø14262  ØØ5Ø37  ØØØØ12                  CLR     @#RESVEC+2

        Ø14266  126727  177762  ØØØØØ4          CMPB    OPT.CP,#4       ;BRANCH IF 11/Ø5 OR 11/2Ø
        Ø14274  ØØ24Ø4                          BLT     REL3
        Ø14276  ØØ4767  16473Ø                  JSR     PC,.PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        Ø143Ø2  Ø16627                          ILLTEST
        Ø143Ø4  ØØØØØØ                          HALT


                                        ;3333333333333 FIRST ADDRESS TO BE RELOCATED 333333333
        Ø143Ø6  Ø1Ø7ØØ                  REL3:   MOV     PC,RØ           ;GET PC
        Ø1431Ø  ØØ574Ø                          TST     -(RØ)           ;RØ CONTAINS THE ADDRESS OF REL3
        Ø14312  Ø1ØØ37  ØØ1Ø1Ø                  MOV     RØ,@#FRSTAD     ;SAVE
        Ø14316  Ø1Ø7ØØ                          MOV     PC,RØ           ;GET CURRENT PC
        Ø1432Ø  1627ØØ  Ø1432Ø                  SUB     #.,RØ           ;SUBTRACT RELOCATION FACTOR
        Ø14324  Ø1ØØ37  ØØ1ØØ4                  MOV     RØ,@#FACTOR     ;SAVE RELOCATION FACTOR
        Ø1433Ø  Ø1Ø7Ø1                          MOV     PC,R1           ;SET NEW SCOPE PTR

                                        ;CHECK STACK OVERFLOW
        Ø14332  Ø13767  177776  ØØØ3Ø6  OVFLW:  MOV     @#PSW,7$        ;SAVE STATUS IN 7$ BELOW
        Ø1434Ø  ØØ5Ø37  177776                  CLR     @#PSW           ;SET KERNEL MODE
        Ø14344  Ø1Ø746                          MOV     PC,-(SP)        ;PUSH CURRENT PC ONTO STACK
        Ø14346  Ø62716  ØØØ136                  ADD     #2$-.,(SP)      ;FORM ADDRESS OF 2$ BELOW
        Ø14352  Ø11637  ØØØØØ4                  MOV     (SP),@#ERRVEC   ;SET ERROR VECTOR
        Ø14356  Ø12737  ØØØ34Ø  ØØØØØ6          MOV     #34Ø,@#ERRVEC+2 ;SET PRIORITY LEVEL 7 ON TRAP
        Ø14364  Ø62716  ØØØØ74                  ADD     #41$-2$,(SP)    ;FORM ADDRESS OF 41$ BELOW
        Ø1437Ø  Ø12637  ØØØØ2Ø                  MOV     (SP)+,@#IOTVEC  ;SET IOT TRAP VECTOR TO 41$
        Ø14374  Ø12746  ØØØ34Ø                  MOV     #34Ø,-(SP)
        Ø144ØØ  Ø11637  ØØØØ22                  MOV     (SP),@#IOTVEC+2 ;SET PRIORITY LEVEL 7 ON IOT TRAP
        Ø144Ø4  Ø1Ø746                          MOV     PC,-(SP)        ;PUSH CURRENT PC ONTO THE STAK
        Ø144Ø6  Ø62716  ØØØØØ6                  ADD     #6,(SP)         ;ADD OFFSET TO INST FOLLOWING RTI
        Ø14412  ØØØØØ2                          RTI                     ;SET PRIORITY LEVEL 7,CLEAR 'T' BIT
                                                                        ;AND EXECUTE FOLLOWING INST NEXT
        Ø14414  Ø127Ø3  ØØØ376                  MOV     #376,R3
        Ø1442Ø  Ø1Ø313                          MOV     R3,(R3)         ;LOAD 376 INTO ADDRESS 376
        Ø14422  Ø1Ø3Ø6                          MOV     R3,SP           ;SET STACK PTR AT BOUNDARY

                                        ;THE BELOW INSTRUCTIONS SHOULD NOT CAUSE AN OVERFLOW TRAP
        Ø14424  ØØ5716                          TST     (SP)            ;BECAUSE TST IS A NON MODIFYING INST
        Ø14426  Ø21666  177776                  CMP     (SP),-2(SP)     ;SO IS COMPARE
        Ø14432  122737  ØØØØØ2  Ø14254          CMPB    #2,@#OPT.CP     ;CHECK IF 11/2Ø OR 11/Ø5
        Ø1444Ø  ØØ2411                          BLT     12$             ;BRANCH IF 11/4Ø OR 11/45
        Ø14442  ØØ14Ø4                          BEQ     11$             ;BRANCH IF 11/2Ø
        Ø14444  Ø12767  ØØØØ14  ØØØ144          MOV     #14,51$         ;CHANGE CHECK WORD IN 51$ IF 11/Ø5
        Ø14452  ØØØ4Ø7                          BR      1Ø$
        Ø14454  Ø12767  ØØØØ34  ØØØ134  11$:    MOV     #34,51$         ;CHANGE CHECK WORD IN 51$ IF 11/2Ø
        Ø14462  ØØØ4Ø3                          BR      1Ø$
        Ø14464  Ø12656                  12$:    MOV     (SP)+,@-(SP)    ;BECAUSE OF ADDRESS MODE 5
        Ø14466  Ø54676  ØØØØØØ                  BIS     -(SP),@(SP)     ;BECAUSE OF ADDRESS MODE 7
        Ø14472  ØØ5Ø66  ØØØØØ4          1Ø$:    CLR     4(SP)           ;BECAUSE DEST ADDRESS IS > 376
        Ø14476  Ø57636  ØØØØØØ                  BIS     @(SP),@(SP)+    ;BECAUSE OF ADDRESS MODE 3
        Ø145Ø2  ØØØ4Ø6                          BR      3$              ;BRANCH OVER NON KERNEL MODE TESTS


                                        ;ERROR SERVICE ROUTINE
        Ø145Ø4  Ø126ØØ                  2$:     MOV     (SP)+,RØ        ;SAVE PC OF INSTRUCTION THAT TRAPPED
        Ø145Ø6  Ø126Ø2                          MOV     (SP)+,R2        ;SAVE PSW
        Ø1451Ø  Ø127Ø6  ØØØ5ØØ                  MOV     #STKPTR,SP      ;SET STACK PTR
        Ø14514  1Ø44ØØ                          HLT                     ;ERROR! AN INSTRUCTION THAT WAS NOT
                                                                        ;SUPPESED TO TRAP TRAPPED
                                                                        ;RØ CONTAINS PC, R2 CONTAINS PSW
        Ø14516  ØØØ45Ø                          BR      6$              ;EXIT TEST
                                        ;THE BELOW INSTRUCTIONS WILL CAUSE A STACK OVERFLOW
                                        ;STACK PTR IS AT 376
        Ø1452Ø  Ø62737  ØØØØ66  ØØØØØ4  3$:     ADD     #4$-2$,@#ERRVEC ;SET ERROR VECTOR TO 4$
        Ø14526  Ø1Ø3Ø6                          MOV     R3,SP           ;SET STACK PTR AT 376
        Ø1453Ø  Ø127Ø2  ØØØØØ1                  MOV     #1,R2
        Ø14534  ØØ5ØØØ                          CLR     RØ
        Ø14536  ØØ5Ø16                          CLR     (SP)            ;SETS BIT Ø IN RØ
```

```
        Ø1454Ø  ØØ63Ø2                          ASL     R2              ;SHIFT INDICATOR BIT
        Ø14542  1Ø5226                          INCB    (SP)+           ;SETS BIT 1 IN RØ
        Ø14544  ØØ63Ø2                          ASL     R2
        Ø14546  Ø6Ø746                          ADD     PC,-(SP)        ;SETS BIT 2 IN RØ
        Ø1455Ø  ØØ63Ø2                          ASL     R2
        Ø14552  ØØØØØ4                          IOT                     ;SETS BIT 3 IN RØ
        Ø14554  ØØ63Ø2                          ASL     R2
        Ø14556  ØØ4767  ØØØØ14                  JSR     PC,4Ø$          ;SETS BIT 4 IN RØ
        Ø14562  ØØ63Ø2                          ASL     R2              ;NOTE! 11/Ø5 WITHOUT ECO # KD11A-ØØØØ5
                                                                        ;DOES NOT SET BIT 4.
        Ø14564  Ø5Ø666  177776                  BIS     SP,-2(SP)       ;SETS BIT 5 IN RØ
        Ø1457Ø  ØØØ4Ø7                          BR      5$

                                        ;PROGRAM WILL TRAP HERE ON OVERFLOW TRAP
        Ø14572  Ø5Ø2ØØ                  4$:     BIS     R2,RØ           ;SET APPROPRIATE BIT IN RØ
        Ø14574  ØØØØØ2                          RTI                     ;RETURN FROM TRAP

        Ø14576  ØØØ2Ø7                  4Ø$:    RTS     PC

        Ø146ØØ  Ø12737  ØØØØ22  ØØØØ2Ø  41$:    MOV     #IOTVEC+2,@#IOTVEC
        Ø146Ø6  ØØØØØ2                          RTI

                                        ;CHECK THAT ABOVE INSRUCTIONS DID TRAP
        Ø1461Ø  Ø127Ø6  ØØØ5ØØ          5$:     MOV     #STKPTR,SP      ;SET STACK PTR
        Ø14614  Ø227ØØ                  5Ø$:    CMP     (PC)+,RØ        ;EACH INSTRUCTION SET A BIT IN RØ
        Ø14616  ØØØØØØ                  51$:    .WORD   Ø               ;CONTAINS CHECK WORD
        Ø1462Ø  ØØ14Ø7                          BEQ     6$              ;RØ= 77 IF 4Ø OR 45,14 IF Ø5,34 IF 2Ø
        Ø14622  1Ø5737  Ø14254                  TSTB    @#OPT.CP        ;CHECK IF 11/Ø5
        Ø14626  ØØ1ØØ3                          BNE     52$             ;BRANCH IF NOT AN 11/Ø5
        Ø1463Ø  Ø227ØØ  ØØØØ34                  CMP     #34,RØ          ;USE ECO KD11A-ØØØØ5 CHECK WORD
        Ø14634  ØØ14Ø1                          BEQ     6$
        Ø14636  1Ø44ØØ                  52$:    HLT

                                        ;EXIT ROUTINE
        Ø1464Ø  Ø127Ø6  ØØØ6ØØ          6$:     MOV     #KPTR,SR        ;SET KERNEL STACK PTR
        Ø14644  Ø12746                          MOV     (PC)+,-(SP)     ;PUSH OLD PSW ONTO STACK
        Ø14646  ØØØØØØ                  7$:     .WORD   Ø               ;CONTAINS SAVED PSW
        Ø1465Ø  Ø1Ø746                          MOV     PC,-(SP)        ;PUSH CURRENT PC ONTO STACK
        Ø14652  Ø62716  ØØØØØ6                  ADD     #6,(SP)         ;ADD OFFSET
        Ø14656  ØØØØØ2                          RTI
        Ø1466Ø  Ø127Ø6  ØØØ5ØØ                  MOV     #STKPTR,SP      ;SET STACK PTR
        Ø14664  Ø12737  ØØØØØ6  ØØØØØ4          MOV     #ERRVEC+2,@#ERRVEC
        Ø14672  1Ø4ØØØ                          SCOPE

                                        ;CHECK THAT ALL RESERVED INSTRUCTIONS TRAP (TO LOCATION 1Ø)
        Ø14674  Ø12737  ØØØØØ2  ØØ1114  RESTRP: MOV     #2,@#SCOPED     ;LIMIT TO TWO ITERATIONS
        Ø147Ø2  Ø1Ø7Ø1                          MOV     PC,R1           ;SET SCOPE POINTER
        Ø147Ø4  Ø127Ø2  Ø15Ø24                  MOV     #5$,R2          ;GET ADDRESS OR RESERVED INSTRUCTION TABLE
        Ø1471Ø  Ø637Ø2  ØØ1ØØ4                  ADD     @#FACTOR,R2
        Ø14714  122737  ØØØØØ4  Ø14254          CMPB    #4,@#OPT.CP     ;ADJUST TABLE ADDRESS IF 11/2Ø, 11/Ø5
        Ø14722  ØØ34Ø2                          BLE     11$             ;5$=11/45, 11/4Ø TABLE, 6$=11/Ø5
        Ø14724  Ø627Ø2  ØØØØ36                  ADD     #6$-5$,R2       ;11/2Ø TABLE
        Ø1473Ø  132737  ØØØØ4Ø  Ø14255  11$:    BITB    #4Ø,@#OPT.CP+1  ;CHECK IF 11/45 FLOATING POINT IS AVAIL.
        Ø14736  ØØ14Ø2                          BEQ     .+6             ;BRANCH IF NOT AVAILABLE
```

```
      014740  005067  000110                      CLR     50$             ;SET TABLE TERMINATOR AT GROUP 7
      014744  012737  015002  000010              MOV     #4$,@#RESVEC    ;SET RESERVED INSTRUCTION TRAP
      014752  063737  001004  000010              ADD     @#FACTOR,@#RESVEC
      014760  012203                      1$:     MOV     (R2)+,R3        ;GET FIRST RESERVED INSTRUCTION
      014762  001454                              BEQ     7$              ;0 TERMINATES THE TABLE
      014764  012204                              MOV     (R2)+,R4        ;GET LAST RESERVED INSTRUCTION IN GROUP
      014766  010317                      2$:     MOV     R3,(PC)         ;EXECUTE RESERVED INSTRUCTION
      014770  000000                      3$:     .WORD   0               ;CONTAINS RESERVED INSTRUCTION
      014772  104400                              HLT                     ;ERROR! INSTRUCTION IN R3
      014774  104400                              HLT                     ;(2$) ABOVE FAILED TO CAUSE A
      014776  104400                              HLT                     ;RESERVED INSTRUCTION TRAP
      015000  000405                              BR      41$
      015002  012716  015014              4$:     MOV     #41$,(SP)       ;ADJUST RETURN PC
      015006  063716  001004                      ADD     @#FACTOR,(SP)   ;TO RETURN TO 41$
      015012  000002                              RTI                     ;RETURN TO 41$
      015014  020304                      41$:    CMP     R3,R4           ;HAS GROUP OF RESERVED INSTRUCTIONS
      015016  001760                              BEQ     1$              ;BEEN EXECUTED
      015020  005203                              INC     R3              ;INCREMENT THIS RESERVED INSTRUCTION
      015022  000761                              BR      2$              ;TO NEXT ONE AND EXECUTE
                                          ;TABLE OF 11/40,11/45 RESERVED INSTRUCTIONS (0 TERMINATES THE TABLE)
      015024  000007                      5$:     7                       ;GROUP 1
      015026  000077                              77                      ;   "
      015030  000210                              210                     ;GROUP 2
      015032  000227                              227                     ;   "
      015034  007000                              7000                    ;GROUP 3
      015036  007777                              7777                    ;   "
      015040  075040                              75040                   ;GROUP 4
      015042  076777                              76777                   ;   "
      015044  106400                              106400                  ;GROUP 5
      015046  106477                              106477                  ;   "
      015050  106700                              106700                  ;GROUP 6
      015052  107777                              107777                  ;   "
      015054  170000                      50$:    170000                  ;GROUP 7        FLOATING POINT
      015056  177777                              177777                  ;   "           INSTRUCTIONS
      015060  000000                              0                       ;0 TERMINATES THE TABLE

                                          ;TABLE OF 11/05, 11/20 RESERVED INSTRUCTIONS (0 TERMINATES THE TABLE)
      015062  000006                      6$:     6                       ;GROUP 1
      015064  000077                              77                      ;   "
      015066  000210                              210                     ;GROUP 2
      015070  000237                              237                     ;   "
      015072  006400                              6400                    ;GROUP 3
      015074  007777                              7777                    ;   "  3
      015076  070000                              70000                   ;GROUP 4
      015100  077777                              77777                   ;   "
      015102  106400                              106400                  ;GROUP 5
      015104  107777                              107777                  ;   "
      015106  170000                              170000                  ;GROUP 6
      015110  177777                              177777                  ;   "
      015112  000000                              0                       ;0 TERMINATES THE TABLE
      015114  012737  000012  000010      7$:     MOV     #RESVEC+2,@#RESVEC      ;RESTORE RESERVED TRAP TO HALT AT 12
      015122  104000                              SCOPE

                                          ;CHECK THAT ALL BITS IN THE PROCESSER STATUS WORD (PSW) CAN BE SET AND
```

```
                                          ;CLEARED.
      015124  013767  177776  000152      PSWCHK: MOV     @#PSW,3$        ;SAVE STATUS
      015132  005037  177776                      CLR     @#PSW           ;CLEAR MODE BITS IN PSW
      015136  005046                              CLR     -(SR)           ;ROUTINE TO CLEAR
      015140  010746                              MOV     PC,-(SP)        ;STATUS WORD (PSW)
      015142  062716  000006                      ADD     #6,(SP)
      015146  000002                              RTI                     ;CLEAR PSW & EXECUTE FOLLOWING INST

      015150  013746  000016                      MOV     @#TBITVEC+2,-(SP)
      015154  012704  177776                      MOV     #PSW,R4         ;LOAD ADDRESS OF PSW INTO R4
      015160  000250                              CLN
      015162  005714                              TST     (R4)            ;CHECK THAT PSW WAS CLEARED
      015164  001401                              BEQ     .+4
      015166  104400                              HLT                     ;ERROR! PSW FAILED TO CLEAR
      015170  113700  014254                      MOVB    @#OPT.CR,R0     ;GET CP TYPE
      015174  016000  016566                      MOV     PSWBIT(0),R0    ;GET BIT MASK FOR TEST R0=THOSE BITS IN
                                                                          ;THE PSW WHICH CAN BE SET/CLEARED.
      015200  005737  014254                      TST     @#OPT.CR        ;CHECK IF MEM MGMT IS AVAILABLE
      015204  100002                              BPL     10$             ;BRANCH IF NOT AVAILABLE
      015206  052700  170000                      BIS     #170000,R0      ;SET BITS 15-12 IF MEM MGMT
      015212  012702  000001              10$:    MOV     #1,R2           ;R2 = TEST BIT
      015216  030200                      1$:     BIT     R2,R0           ;CHECK IF BIT CAN BE SET/CLEARED
      015220  001423                              BEQ     2$
      015222  005037  000016                      CLR     @#TBITVEC+2
      015226  030227  000020                      BIT     R2,#20          ;CHECK IF TEST WILL SET 'T' BIT
      015232  001403                              BEQ     20$
      015234  012737  000002  000016              MOV     #RTI,@#TBITVEC+2;SET RTI INTO RETURN
      015242  005014                      20$:    CLR     (R4)            ;CLEAR PSW
      015244  050214                              BIS     R2,(R4)         ;SET R2 INTO PSW
      015246  011403                              MOV     (R4),R3         ;GET BIT
      015250  020203                              CMP     R2,R3           ;CHECK THAT BIT WAS SET IN PSW
      015252  001401                              BEQ     .+4
      015254  104400                              HLT                     ;ERROR! BIT IN R2 FAILED TO SET IN PSW
      015256  000244                              CLZ                     ;CLEAR Z BIT
      015260  040214                              BIC     R2,(R4)         ;CLEAR BIT IN PSW
      015262  011403                              MOV     (R4),R3         ;GET PSW RESULT
      015264  001401                              BEQ     2$              ;BRANCH IF BIC ABOVE CLEARED BIT IN PSW
      015266  104400                              HLT                     ;ERROR! BIT IN R2 FAILED TO CLEAR IN PSW
      015270  006302                      2$:     ASL     R2              ;SHIFT TEST BIT
      015272  103351                              BCC     1$              ;BRANCH IF ALL BITS NOT TESTED
      015274  005014                              CLR     (R4)            ;CLEAR STATUS
      015276  012637  000016                      MOV     (SP)+,@#TBITVEC+2       ;RESTORE T BIT RETURN
      015302  012746                              MOV     (PC)+,-(SP)     ;PUSH ORIGINAL STATUS ON STACK
      015304  000000                      3$:     .WORD   0               ;CONTAINS ORIGINAL PSW
      015306  010746                              MOV     PC,-(SP)        ;SET RETURN PC
      015310  062716  000006                      ADD     #6,(SP)
      015314  000002                              RTI                     ;RETURN
      015316  104000                      4$:     SCOPE

      015320  013704  177776                      MOV     @#PSW,R4                ;SAVE PSW IN R4
      015324  010446                              MOV     R4,-(SP)                ;PUSH R4 ONTO STACK
      015326  112716  000300                      MOVB    #300,(SP)               ;SET PRIORITY LEVEL 6 AND
      015332  010746                              MOV     PC,-(SP)                ;CLEAR 'T' BIT AND EXECUTE
      015334  062716  000006                      ADD     #6,(SP)                 ;INSTRUCTION FOLLOWING RTI
```

```
        015340  000002                          RTI

                                        ;CHECK THAT ALL BITS IN THE CURRENT STACK PTR CAN BE SET/CLEARED
        015342  010603                  CHKSP:  MOV     SP,R3           ;SAVE STACK PTR
        015344  000257                          CCC
        015346  112706  000377                  MOVB    #377,SP         ;SET STACK PTR = -1
        015352  006006                  1$:     ROR     SP              ;ROTATE 0 BIT THROUGH ALL BIT
        015354  103776                          BCS     1$              ;BIT POSITIONS
        015356  005206                          INC     SP              ;SHOULD INCREMENT SP TO 0
        015360  001403                          BEQ     2$
        015362  010602                          MOV     SP,R2           ;SAVE ERROR STACK PTR
        015364  010306                          MOV     R3,SP           ;SET STACK PTR FOR TRAP
        015366  104400                          HLT                     ;ERROR!

        015370  010306                  2$:     MOV     R3,SP           ;RESTORE ORIGINAL STACK PTR

                                        ;CHECK BYTE OPERATIONS USING THE STACK
        015372  010600                  SPCHK:  MOV     SP,R0           ;SAVE STACK PTR
        015374  010003                          MOV     R0,R3
        015376  005043                          CLR     -(R3)
        015400  112746  177777                  MOVB    #-1,-(SP)       ;(SP) = 377
        015404  022713  000377                  CMP     #377,(R3)       ;CHECK THAT ONLY EVEN BYTE WAS AFFECTED
        015410  001002                          BNE     1$
        015412  020306                          CMP     R3,SP           ;CHECK AUTO-DEC
        015414  001401                          BEQ     .+4
        015416  104400                  1$:     HLT

        015420  105226                          INCB    (SP)+
        015422  005723                          TST     (R3)+           ;CHECK RESULT
        015424  001002                          BNE     2$
        015426  020006                          CMP     R0,SP           ;CHECK AUTO-INC
        015430  001401                          BEQ     .+4
        015432  104400                  2$:     HLT

        015434  005143                          COM     -(R3)           ;(R3)=177777
        015436  144613                          BICB    -(SP),(R3)
        015440  022713  177400                  CMP     #177400,(R3)    ;CHECK RESULT
        015444  001002                          BNE     3$
        015446  020603                          CMP     SP,R3
        015450  001401                          BEQ     .+4
        015452  104400                  3$:     HLT

        015454  132627  000377                  BITB    (SP)+,#377
        015460  001002                          BNE     4$
        015462  020600                          CMP     SP,R0
        015464  001401                          BEQ     .+4
        015466  104400                  4$:     HLT

        015470  012746  000001                  MOV     #1,-(SP)
        015474  062706  000002                  ADD     #2,SP
        015500  012702  177401                  MOV     #177401,R2
        015504  120246                          CMPB    R2,-(SP)
        015506  001004                          BNE     5$
        015510  122602                          CMPB    (SP)+,R2
```

```
        015512  001002                          BNE     5$
        015514  020006                          CMP     R0,SP
        015516  001401                          BEQ     .+4
        015520  104400                  5$:     HLT
        015522  010446                          MOV     R4,-(SP)        ;RESTORE ORIGINAL PSW TO STACK
        015524  010746                          MOV     PC,-(SP)
        015526  062716  000006                  ADD     #6,(SP)
        015532  000002                          RTI
        015534  104000                          SCOPE

                                        ;CHECK THAT 'C' BIT SETS/CLEARS PROPERLY
        015536  012727  177776          CBIT:   MOV     #177776,(PC)+   ;LOAD CONSTANT
        015542  000000                  1$:     .WORD   0
        015544  010700                          MOV     PC,R0           ;GET CURRENT PC
        015546  162700  000004                  SUB     #4,R0           ;POINT R0 TO 1$ ABOVE
        015552  005520                  2$:     ADC     (R0)+           ;ADD 'C' BIT TO 1$ ABOVE
        015554  006340                          ASL     -(R0)           ;SHIFT 1$
        015556  102375                          BVC     2$              ;UNTIL 'V' BIT SETS
        015560  022767  077776  177754          CMP     #077776,1$      ;CHECK RESULT
        015566  001401                          BEQ     .+4
        015570  104400                          HLT                     ;ERROR! INCORRECT RESULT IN 1$ ABOVE
                                                                        ;R0=ADDRESS OF DATA

                                        ;CHECK THAT CONDITION CODES ARE SET PROPERLY WHEN A NUMBER (CURRENT PC)
                                        ;AND THAT NUMBER +1 ARE COMPARED, AND VICE VERSA.
        015572  010700                  CMPN:   MOV     PC,R0           ;GET CURRENT PC
        015574  010002                          MOV     R0,R2           ;SAVE IN R2
        015576  005202                          INC     R2              ;MAKE R2 = R0+1
        015600  000277                          SCC
        015602  000251                          +CLC!CLN                ;CLEAR C & N BITS
        015604  020002                          CMP     R0,R2           ;COMPARE # WITH #+1
        015606  103003                          BCC     1$              ;CARRY BIT SHOULD SET
        015610  102402                          BVS     1$              ;V BIT SHOULD CLEAR
        015612  001401                          BEQ     1$              ;Z BIT SHOULD CLEAR
        015614  100401                          BMI     .+4             ;N BIT SHOULD SET
        015616  104400                  1$:     HLT                     ;ERROR! COMPARE # WITH #+1 FAILED TO
                                                                        ;SET CONDITION CODES IN PSW CORRECTLY

        015620  000277                          SCC                     ;SET CONDITION CODES IN PSW
        015622  120200                          CMPB    R2,R0           ;COMPARE #+1 WITH #
        015624  103403                          BCS     2$              ;C BIT SHOULD CLEAR
        015626  102402                          BVS     2$              ;V BIT SHOULD CLEAR
        015630  001401                          BEQ     2$              ;Z BIT SHOULD CLEAR
        015632  100001                          BPL     .+4             ;N BIT SHOLD CLEAR
        015634  104400                  2$:     HLT                     ;ERROR! COMPARE #+1 WITH # FAILED TO SET
                                                                        ;CONDITION CODES IN PSW CORRECTLY

                                        ;24 NOP (240) INSTRUCTIONS FOLLOW. THESE NOPS MAY
                                        ;BE CHANGED TO TEST CODE IF THE NEED ARISES. THE TEST CODE SHOULD
                                        ;BE POSITION INDEPENDENT AND SHOULD RUN WHEN RELOCATED BY THE PROGRAM.
        015636  000240                          NOP
        015640  000240                          NOP
        015642  000240                          NOP
        015644  000240                          NOP
        015646  000240                          NOP
        015650  000240                          NOP
        015652  000240                          NOP
        015654  000240                          NOP
        015656  000240                          NOP
        015660  000240                          NOP
        015662  000240                          NOP
        015664  000240                          NOP
        015666  000240                          NOP
        015670  000240                          NOP
        015672  000240                          NOP
        015674  000240                          NOP
        015676  000240                          NOP
        015700  000240                          NOP
        015702  000240                          NOP
        015704  000240                          NOP
        015706  000240                          NOP
        015710  000240                          NOP
        015712  000240                          NOP
        015714  000240                          NOP
        015716  104000                          SCOPE

        015720  010702                          MOV     PC,R2
        015722  062702  000012                  ADD     #12,R2
        015726  012707  021132                  MOV     #RELOC,PC       ;GO RELOCATE PROGRAM CODE
        015732  000240                          NOP                     ;PROGRAM RETURNS HERE+2
                                        ;3333333333333 LAST ADDRESS OF CODE TO BE RELOCATED 33333333333


                                        ;CHECK TTY INTERRUPT,
        015734  005037  001004          TTYCHK: CLR     @#FACTOR
        015740  010701                          MOV     PC,R1
        015742  032737  000100  177564          BIT     #100,@#TPS      ;CHECK IF TTY IS READY
        015750  001374                          BNE     .-6
        015752  012737  016026  000064          MOV     #3$,@#TPVEC     ;SET TTY INTERRUPT VECTOR
        015760  012737  000200  000066          MOV     #200,@#TPVEC+2  ;PRIORITY LEVEL 4 ON INTERRUPT
        015766  012767  016064  000064          MOV     #NULLS,MSG      ;ADDRESS OF MESSAGE TO BE TYPPED
        015774  117737  000060  177566          MOVB    @MSG,@#TPB      ;TYPE FIRST CHARACTER OF MESSAGE
        016002  105737  177564                  TSTB    @#TPS
        016006  100375                          BPL     .-4
        016010  006237  177564                  ASR     @#TPS           ;SET IE BIT IN TTY CSR REG
        016014  000001                          WAIT                    ;WAIT FOR FIRST INTERRUPT
        016016  000424                          BR      KW11
        016020  006337  177564          2$:     ASL     @#TPS           ;CLEAR IE BIT
        016024  000002                          RTI

        016026  122777  000012  000024  3$:     CMPB    #12,@MSG        ;BRANCH IF CHAR IS NOT <LF>
        016034  001004                          BNE     4$
        016036  004767  163170                  JSR     PC,$PRINT       ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        016042  001744                          $CRLF
        016044  000404                          BR      5$
        016046  117737  000006  177566  4$:     MOVB    @MSG,@#TPB      ;TYPE CHARACTER
        016054  001761                          BEQ     2$              ;BRANCH IF TERMINATOR
        016056  005227                  5$:     INC     (PC)+           ;SET MSG TO NEXT CHAR ADDRESS
```

```
        016060  000000                  MSG:    .WORD   0               ;CONTAINS ADDRESS OF CHAR TO BE TYPED
        016062  000002                          RTI                     ;RETURN
        016064  020015  000015          NULLS:  .ASCIZ  <15><40><15>
                                                .EVEN

                                        ;ROUTINE TO TURN ON KW11-L LINE CLOCK IF AVAILABLE
        016070  012737  000002  000006  KW11:   MOV     #RTI,@#ERRVEC+2         ;SET UP DIRECT RTI ON TRAP
        016076  012737  016232  000100          MOV     #4$,@#LKVEC             ;LOAD INTERRUPT VECTOR
        016104  012737  000300  000102          MOV     #300,@#LKVEC+2          ;SET PRIORITY LEVEL 6 ON INT,
        016112  000262                          SEV                             ;SET TIME OUT INDICATOR
        016114  052737  000100  177546          BIS     #100,@#LKS              ;SET INTERRUPT ENABLE
        016122  102446                          BVS     5$                      ;SKIP PRIORITY ARBITRATION TEST
                                                                                ;BELOW IF NO KW11-L

                                        ;ROUTINE TO CHECK PRIORITY ARBITRATION LOGIC
                                        ;THE BELOW TEST WILL INHIBIT INTERRUPTS ON LEVEL 6 AND ABOVE (LOCKING
                                        ;OUT THE LINE CLOCK) AND THEN SET UP THE TTY TO INTERRUPT, NEXT THE
                                        ;PRIORITY LEVEL WILL BE SET TO 0 ALLOWING INTERRUPTS IN WHICH CASE
                                        ;THE LINE CLOCK (AT LEVEL 6) SHOULD INTERRUPT BEFORE THE TTY (AT LEVEL 4),

        016124  132737  000020  177776          BITB    #20,@#PSW               ;CHECK IF 'T' BIT IS SET
        016132  001042                          BNE     5$                      ;DO NOT DO TEST IF SET
        016134  112737  000300  177776          MOVB    #300,@#PSW              ;SET PRIORITY LEVEL = 6
        016142  013727  000064                  MOV     @#TRVEC,(PC)+           ;SAVE TTY INTERRUPT VECTOR
        016146  000000                  1$:     .WORD   0                       ;CONTAINS CURRENT TTY VECTOR
        016150  105737  177564                  TSTB    @#TPS                   ;CHECK IF READY
        016154  100375                          BPL     .-4                     ;WAIT FOR TTY TO BECOME READY
        016156  012737  016202  000064          MOV     #2$,@#TRVEC             ;SET NEW VECTOR
        016164  005227                          INC     (PC)+                   ;STALL WAITING FOR LINE CLOCK
        016166  000000                          .WORD   0                       ;TO BE READY
        016170  012737  016206  000100          MOV     #3$,@#LKVEC             ;SET LINE CLOCK VECTOR
        016176  105037  177776                  CLRB    @#PSW                   ;SET PRIORITY LEVEL 0
        016202  104400                  2$:     HLT                             ;ERROR! EITHER TTY INTERRUPTED
                                        ;BEFORE THE LINE CLOCK OR BOTH FAILED TO INTERRUPT
        016204  000415                          BR      5$                      ;EXIT TEST
        016206  016737  177734  000064  3$:     MOV     1$,@#TPVEC              ;RESTORE TTY VECTOR
        016214  012737  016232  000100          MOV     #4$,@#LKVEC             ;SET LINE CLOCK VECTOR
        016222  105037  177776                  CLRB    @#PSW                   ;RESTORE PRIORITY LEVEL 0
        016226  012716  016240                  MOV     #5$,(SP)                ;SET RETURN ADDRESS TO 5$ BELOW

        016232  005267  162540          4$:     INC     TICKS                   ;INCREMENT TICK COUNT
        016236  000002                          RTI                             ;RETURN

        016240  005037  000006          5$:     CLR     @#ERRVEC+2              ;RESTORE ERROR TRAP TO HALT AT 6

        016244  000240                  END:    NOP
        016246  005037  177776          END1:   CLR     @#PSW           ;CLEAR MODE BITS IN PSW
        016252  005046                          CLR     -(SP)           ;CLEAR PSW
        016254  012746  016262                  MOV     #.+6,-(SP)
        016260  000002                          RTI                     ;GO TO NEXT INST WITH PSW=0
        016262  012706  000600                  MOV     #KPTR,SP        ;SET KERNEL STACK PTR (NOT APPLICABLE
                                                                        ;FOR 11/20,11/05 CP'S)
        016266  032737  000100  177564          BIT     #100,@#TPS      ;CHECK IF OUTPUT DEVICE IS BUSY
        016274  001374                          BNE     .-6             ;IS AVAILABLE
        016276  105737  177570                  TSTB    @#SWR              ;DELETE END OF PASS TYPE OUT IF SW7=0
        016302  100020                          BPL     1$                 ;BRANCH IF SW7 IS DOWN
        016304  016702  162470                  MOV     ICNT,R2            ;GET PASS COUNT
        016310  004767  163014                  JSR     PC,$FORM0          ;GO TO FORMAT ROUTINE
        016314  012702  001664                  MOV     #DIGITS+2,R2       ;GET ASCII VALUES
        016320  012703  001702                  MOV     #PASSES,R3         ;AND MOVE THEM INTO MESSAGE
        016324  012223                          MOV     (R2)+,(R3)+
        016326  012223                          MOV     (R2)+,(R3)+
        016330  012737  001672  016060          MOV     #PASCNT,@#MSG      ;PASS MESSAGE ADRS TO TELETYPE SERVICE
        016336  052737  000100  177564          BIS     #100,@#TPS         ;SET IE BIT
        016344  012737  000610  000024  1$:     MOV     #PDWN,@#PFVEC      ;ENABLE POWER FAIL TRAP
        016352  012737  000340  000026          MOV     #340,@#PFVEC+2     ;PRIORITY 7 ON POWER FAIL
        016360  005267  162414                  INC     ICNT
        016364  116700  175664                  MOVB    OPT.CP,R0          ;GET CP TYPE
        016370  026067  016572  162402          CMP     PASTAB(R0),ICNT    ;CHECK IF END OF TEST
        016376  001002                          BNE     2$                 ;BRANCH IF NOT AT END
        016400  000167  000060                  JMP     DONE
        016404  016702  162370          2$:     MOV     ICNT,R2            ;GET PASS COUNT
        016410  006302                          ASL     R2
        016412  046002  016562                  BIC     CPPASS(0),R2       ;LIMIT PASS COUNT TO 0-6
        016416  005037  000016                  CLR     @#16               ;CLEAR T BIT TRAP ADDRESS
        016422  012737  000040  001122          MOV     #40,@#SCOPEF+2     ;SET ITERATION COUNT = 40
        016430  016216  016556                  MOV     PSWTAB(2),(SP)     ;PUSH NEXT PASS PSW ON STACK
        016434  032716  000020                  BIT     #20,(SP)           ;WILL 'T' BIT BE SET ON NEXT PASS?
        016440  001406                          BEQ     3$                 ;BRANCH IF NOT
        016442  012737  000002  001122          MOV     #2,@#SCOPEF+2      ;SET ITERATION COUNT = 2 FOR 'T' BIT
        016450  016737  000006  000016          MOV     RTI1,@#16          ;SET 'T' BIT TRAP TO RETURN VIA 16
        016456  012746  002230          3$:     MOV     #START2,-(SP)      ;RESART PROGRAM AT START2
        016462  000002                  RTI1:   RTI                        ;RESTART PROGRAM AT START2 WITH NEW PSW
                                                                           ;(FROM TABLE BELOW) NOTE: THE RTI IS
                                                                           ;CHANGED TO AN RTT IF NOT AN 11/05,11/20

                                        ;ROUTINE TO SET UP MEMORY MANAGEMENT TO RELOCATE PROGRAM CODE ABOVE 28K

        016464  032737  000100  177564  DONE:   BIT     #100,@#TPS         ;WAIT FOR TTY OUTPUT TO FINISH
        016472  001374                          BNE     DONE
        016474  105737  177564                  TSTB    @#TPS              ;WAIT FOR LAST CHARACTER TO BE PRINTED
        016500  100375                          BPL     .-4
        016502  005027                          CLR     (PC)+
        016504  000000                  1$:     .WORD   0
        016506  005267  177772          2$:     INC     1$                 ;DELAY WAITING FOR TELETYPE TO FINISH
        016512  001375                          BNE     2$                 ;TYPING CHARACTER BEFORE ISSUING RESET
        016514  000005                          RESET
        016516  105737  177570                  TSTB    @#SWR
        016522  100003                          BPL     3$
        016524  004767  162502                  JSR     PC,.PRINT          ;PRINT MESSAGE BEGINING AT FOLLOWING ADRS
        016530  016716                          ENDMSG
        016532  013702  000042          3$:     MOV     @#42,R2            ;CHECK DDP/ACT11 MONITOR HOOK
        016536  001405                          BEQ     DONE1
        016540  000005                          RESET
        016542  004712                  LOGICAL:JSR     PC,(R2)            ;GO TO DDP/ACT11 MONITOR VIA 42
        016544  000240                          NOP
        016546  000240                          NOP
        016550  000240                          NOP
```

```
        016552  000137  002224          DONE1:  JMP     @#START0                ;RESTART PROGRAM

                                        ;THE BELOW TABLE RERRESENTS THE 'NEW' PSW SET BY THE PROGRAM ON
                                        ;SUCCESSIVE PASSES.
                                        ;NOTE THE BELOW TABLE MAY BE MODIFIED TO CAUSE THE PROGRAM TO RUN
                                        ;UNDER USER DEFINED PARAMETERS BY PATCHING IN THE DESIRED PASS PARAMETER
                                        ;FOR EXAMPLE TO CAUSE THE PROGRAM TO RUN WITHOUT SETTING THE 'T' BIT
                                        ;IN ALL PASSES PATCH OUT THE 'T' BIT IN THE TABLE.
        016556  000000                  PSWTAB: 000000                  ;ALL 11 FAMILY CP'S
        016560  000020                          000020

                                        ;THE BELOW TABLE IS THE 'BIT MASK' USED TO DETERMINE THE INDEX VALUE
                                        ;NEEDED TO SET THE 'NEW' PSW.
        016562  177774                  CPPASS: 177774                  ;11/05
        016564  177774                          177774                  ;11/20

                                        ;THE BELOW TABLE RERRESENTS THOSE BITS IN THE CP WHICH CAN BE SET/CLEARED
        016566  000377                  PSWBIT: 000377                  ;11/05
        016570  000377                          000377                  ;11/20

                                        ;THE BELOW TABLE CONTAINS THE # OF PASSES REQUIRED TO COMPLETE TEST
        016572  000002                  PASTAB: .WORD   2               ;11/05
        016574  000002                          .WORD   2               ;11/20
                                        ;MESSAGES
        016576  005015  047514  020127  MSG1:   .ASCIZ  <15><12>'LOW LIMIT?'
        016604  044514  044515  037524
        016612     000
        016613     110  043511  020110  MSG2:   .ASCIZ  'HIGH LIMIT?'
        016620  044514  044515  037524
        016626     000
        016627     015  052012  044510  ILLTEST:.ASCIZ  <15><12>'THIS TEST INVALID FOR 11/40-11/45 PLEASE RUN DCQKC'<15><12>
        016634  020123  042524  052123
        016642  044440  053116  046101
        016650  042111  043040  051117
        016656  030440  027461  030064
        016664  030455  027461  032464
        016672  050040  042514  051501
        016700  020105  052522  020116
        016706  041504  045521  006503
        016714  000012
        016716  005015  042040  050532  ENDMSG: .ASCIZ  <15><12>' DZQKC DONE'
        016724  041513  042040  047117
        016732  000105
                000001                          .END
```

```
ADCB2   ØØ4564          ADCB5   ØØ5374          ADCB6   ØØ6Ø62          ADCB7   ØØ671Ø
ADCØ    ØØ2524          ADC1    ØØ34ØØ          ADC2    ØØ4374          ADC5    ØØ52Ø2
ADC6    ØØ5672          ADC7    ØØ66Ø4          ADDØ    ØØ7376          ADD1    ØØ76Ø2
ADD1A   Ø1ØØ26          ADD1B   Ø1ØØ44          ADD2    Ø1Ø432          ADD3    Ø111Ø6
ADD6    Ø1145Ø          ADD7    Ø12112          ASLB1   ØØ3742          ASLB1A  ØØ4166
ASLB3   ØØ5364          ASLB4   ØØ467Ø          ASLB6   ØØ6Ø44          ASLB7   ØØ7ØØ6
ASLØ    ØØ2646          ASL1    ØØ3554          ASL3    ØØ5316          ASL4    ØØ4466
ASL6    ØØ5642          ASL7    ØØ6432          ASRB1   ØØ4Ø36          ASRB1A  ØØ4Ø52
ASRB2   ØØ4634          ASRB2A  ØØ4652          ASRB5   ØØ5324          ASRB6   ØØ6162
ASRB7   ØØ7Ø24          ASRØ    ØØ2674          ASR1    ØØ3442          ASR2    ØØ441Ø
ASR3    ØØ51Ø2          ASR6    ØØ5524          ASR7    ØØ6466          BELL    ØØ1747
BICB1   Ø1Ø22Ø          BICB1A  Ø1Ø242          BICØ    ØØ731Ø          BIC1    ØØ7724
BIC2    Ø1Ø522          BIC3    Ø1112Ø          BIC7    Ø1257Ø          BINB7   Ø12336
BIN1    Ø1Ø4ØØ          BISB1   Ø1Ø2Ø6          BISØ    ØØ7266          BISØA   ØØ7344
BIS1    ØØ7712          BIS2    Ø1Ø46Ø          BIS2A   Ø1Ø562          BIS7    Ø1253Ø
BITB1   Ø1Ø176          BITB2   Ø1Ø736          BITB3   Ø1127Ø          BITB6   Ø11654
BIT1    ØØ764Ø          BIT13 = Ø2ØØØØ          BIT14 = Ø4ØØØØ          BIT15 = 1ØØØØØ
BIT2    Ø1Ø546          BIT6  = ØØØ1ØØ          BIT8  = ØØØ4ØØ          BPTVEC= ØØØØ14
C     = ØØØØØ1          CBIT    Ø15536          CCØ     ØØ234Ø          CC1     ØØ2354
CC2     ØØ237Ø          CC3     ØØ24Ø2          CC4     ØØ2416          CHKSP   Ø15342
CLRØ    ØØ2442          CMPB1   Ø1Ø152          CMPB2   Ø1Ø722          CMPB3   Ø113Ø2
CMPN    Ø15572          CMPØ    ØØ717Ø          CMPØA   ØØ7432          CMP1    ØØ7622
CMP1A   ØØ7742          CMP2    Ø1Ø45Ø          CMP7    Ø12Ø46          COMB1   ØØ4Ø2Ø
COMB1A  ØØ42ØØ          COMB2   ØØ4546          COMB5   ØØ53Ø2          COMB6   ØØ6114
COMB7   ØØ6724          COMØ    ØØ25Ø6          COM1    ØØ3566          COM3    ØØ516Ø
COM4    ØØ4276          COM6    ØØ551Ø          COM7    ØØ6556          CPCHK   Ø142Ø4
CPPASS  Ø16562          DBINB7  Ø12332          DBIN7   Ø11746          DDATA   Ø11312
DDATAB  Ø11742          DECB1   ØØ377Ø          DECB1A  ØØ412Ø          DECB2   ØØ47Ø2
DECB5   ØØ5442          DECB6A  ØØ6214          DECB7   ØØ6772          DECØ    ØØ2566
DEC1    ØØ336Ø          DEC1A   ØØ363Ø          DEC2    ØØ4444          DEC5    ØØ5134
DEC6    ØØ5656          DEC7    ØØ645Ø          DIGITS  ØØ1662          DIGTAB  ØØ1652
DISPLA= 17757Ø          DONE    Ø16464          DONE1   Ø16552          EMTVEC= ØØØØ3Ø
EMT1    Ø13724          EMT1B   Ø14ØØ4          EMT1C   Ø14Ø1Ø          EMT1D   Ø14Ø22
END     Ø16244          ENDMSG  Ø16716          END1    Ø16246          ERROR   ØØ1412
ERRPC   ØØ17Ø7          ERRPCØ  ØØ1721          ERRVEC= ØØØØØ4          FACTOR  ØØ1ØØ4
FORXOR  ØØØ74Ø          FPEVEC= ØØØ244          FRSTAD  ØØ1Ø1Ø          FRSTME  ØØ1Ø12
GSTST   ØØ316Ø          HLT   = 1Ø44ØØ          ICNT    ØØ1ØØØ          ILLTES  Ø16627
INCB1   ØØ37Ø4          INCB2   ØØ4776          INCB3   ØØ5334          INCB6   ØØ6Ø26
INCB6A  ØØ6176          INCB7   ØØ6756          INCØ    ØØ26Ø6          INC1    ØØ3474
INC3    ØØ517Ø          INC4    ØØ4326          INC6    ØØ562Ø          INC7    ØØ6616
IOTVEC= ØØØØ2Ø          IOT1    Ø13624          IOT1A   Ø13632          IOT1B   Ø13646
JMP1    Ø13Ø52          JMP3    Ø13122          JMP4    Ø1316Ø          JMP5    Ø13224
JMP6    Ø13244          JMP7    Ø133Ø2          JSRTST  Ø133Ø6          JSR1    Ø1337Ø
JSR1A   Ø13372          JSR3    Ø13456          JSR3A   Ø1346Ø          JSR4    Ø13534
JSR4A   Ø13536          JSR6A   Ø1357Ø          KPTR  = ØØØ6ØØ          KW11    Ø16Ø7Ø
LKS   = 177546          LKVEC = ØØØ1ØØ          LOGICA  Ø16542          LPB   = 177516
LPS   = 177514          LSTMEM  ØØ214Ø          MMVEC = ØØØ25Ø          MOVB1   Ø1Ø126
MOVØ    ØØ712Ø          MOVØA   ØØ715Ø          MOV1    Ø1ØØ12          MOV7    Ø12Ø16
MSG     Ø16Ø6Ø          MSG1    Ø16576          MSG2    Ø16613          N     = ØØØØ1Ø
NEGB1   ØØ3754          NEGB4   ØØ4726          NEGB6   ØØ6132          NEGB7   ØØ7Ø4Ø
NEGØ    ØØ2626          NEG1    ØØ362Ø          NEG2    ØØ435Ø          NEG5    ØØ5146
NEG6    ØØ5544          NEG7    ØØ6524          NULL  = ØØØØØØ          NULLS   Ø16Ø64
OAERR   Ø12616          OPT.CP  Ø14254          OVFLW   Ø14332          PASCNT  ØØ1672
PASSES  ØØ17Ø2          PASTAB  Ø16572          PC    =%ØØØØØ7          PDWN    ØØØ61Ø
```

```
PFAIL   ØØØ656          PFVEC = ØØØØ24          PIRQ  = 177772          PIRVEC= ØØØ24Ø
PRTY4 = ØØØ2ØØ          PRTY6 = ØØØ3ØØ          PRTY7 = ØØØ34Ø          PSW   = 177776
PSWBIT  Ø16566          PSWCHK  Ø15124          PSWTAB  Ø16556          PUP     ØØØ62Ø
RECD    ØØ1752          RELOC   ØØ1132          RELR1   ØØ1ØØ6          RELØ    ØØ2272
REL1    ØØ63Ø2          REL2    Ø12274          REL3    Ø143Ø6          RESTRP  Ø14674
RESVEC= ØØØØ1Ø          RETPC   ØØ123Ø          ROLB1   ØØ3716          ROLB2   ØØ462Ø
ROLB3   ØØ541Ø          ROLB6   ØØ6146          ROLB6A  ØØ6242          ROLB7   ØØ7Ø56
ROLØ    ØØ266Ø          ROL1    ØØ3414          ROL1A   ØØ3426          ROL3    ØØ5216
ROL4    ØØ4424          ROL6    ØØ5474          ROL7    ØØ6634          RORB1   ØØ4ØØ4
RORB1A  ØØ4Ø74          RORB4   ØØ46Ø2          RORB5   ØØ5352          RORB6   ØØ61ØØ
RORB7   ØØ6742          RORØ    ØØ2544          ROR1    ØØ3344          ROR1A   ØØ3456
ROR2    ØØ4312          ROR5    ØØ5Ø7Ø          ROR6    ØØ5564          ROR7    ØØ65Ø6
RTI1    Ø16462          RØ    =%ØØØØØØ          R1    =%ØØØØØ1          R1Ø   =%ØØØØØØ
R11   =%ØØØØØ1          R12   =%ØØØØØ2          R13   =%ØØØØØ3          R14   =%ØØØØØ4
R15   =%ØØØØØ5          R2    =%ØØØØØ2          R3    =%ØØØØØ3          R4    =%ØØØØØ4
R5    =%ØØØØØ5          SBCB1   ØØ373Ø          SBCB3   ØØ543Ø          SBCB4   ØØ4714
SBCB6   ØØ6ØØ2          SBCB7   ØØ667Ø          SBCØ    ØØ271Ø          SBC1    ØØ351Ø
SBC1A   ØØ3526          SBC5    ØØ523Ø          SBC6    ØØ56ØØ          SBC7    ØØ6412
SBINB7  Ø12322          SBIN7   Ø11744          SCOPE = 1Ø4ØØØ          SCOPEA  ØØ1Ø16
SCOPEB  ØØ1Ø46          SCOPEC  ØØ11Ø2          SCOPED  ØØ1114          SCOPEE  ØØ1126
SCOPEF  ØØ112Ø          SDATA   Ø1131Ø          SDATAB  Ø1174Ø          SLR   = 177774
SP    =%ØØØØØ6          SPCHK   Ø15372          SRØ   = 177572          START   ØØ2Ø66
START1  ØØ216Ø          START2  ØØ223Ø          START3  ØØ2224          STATUS  ØØ1714
STKPTR= ØØØ5ØØ          SUBØ    ØØ7136          SUB1    ØØ766Ø          SUB1A   ØØ7766
SUB1B   Ø1ØØØ2          SUB2    Ø1Ø476          SUB2A   Ø1Ø616          SUB3    Ø11Ø32
SUB3A   Ø11Ø54          SUB6    Ø1147Ø          SUB7    Ø12Ø66          SWABØ   ØØ2726
SWAB1   ØØ4134          SWAB2   ØØ434Ø          SWAB4   ØØ476Ø          SWAB6   ØØ6226
SWAB7   ØØ654Ø          SWR   = 17757Ø          T     = ØØØØ2Ø          TBITVE= ØØØØ14
TICKS   ØØØ776          TKB   = 177562          TKS   = 17756Ø          TPB   = 177566
TPS   = 177564          TPVEC = ØØØØ64          TRAPVE= ØØØØ34          TRAP1   Ø14112
TRAP1C  Ø14136          TRTVEC= ØØØØ14          TSTB1   ØØ4146          TSTB2   ØØ4736
TSTB2A  ØØ4746          TSTB6   ØØ5754          TSTØ    ØØ2464          TST1    ØØ36Ø4
TST2    ØØ4266          TST6    ØØ6256          TTYCHK  Ø15734          UBM6    ØØ6262
UBREAK= 17777Ø          UB7     Ø11754          UWM6    ØØ545Ø          UWM7    ØØ6332
UW7     ØØ6336          V     = ØØØØØ2          Z     = ØØØØØ4          $CRLF   ØØ1744
$FILLS  ØØ1ØØ2          $FORMØ  ØØ133Ø          $RESTR  ØØØ24Ø          $SAVR   ØØØ214
.PRINT  ØØ1232          .     = Ø16734


 ERRORS DETECTED:  Ø/2

 *,DZQKCE<DZQKCE,P11
 RUN-TIME: 12 22 Ø SECONDS
 CORE USED:  8K
```

digital

MAINDEC
CHANGE ORDER

ORIGINATOR Barry G. Irrgang
TEL EXT X3616 DATE 11/18/75
DISC PROJ NO. V98 05169
COST CENTER NO. 301 MC Ø163

MCO NO. MD-11-DZQKC-E1/Ø1
SHEET 1 OF 1
DATE RECEIVED 4-Feb-76
FIRST ISSUE ________
FINAL ISSUE ________

PROBLEM Arbitration logic test does not work properly with console DL11 over 600 baud and KW11-L present.

PGM TO BE CHANGED
DZQKC

DISP CODE

CORRECTION
1. Delay test long enough to ensure KW11-L has interrupt up.
2. Raise PSW of error routing to ensure error will be reported properly if one occurs.

OPTIONS AFFECTED

BREAK-IN/EFFECTIVITY

PRODUCT LINES AFFECTED
PDP-11

ITEM NO.	DOCUMENT/PART NO.	OLD REV	NEW REV	DISP CODE	DESCRIPTION OF CHANGE
1.	MAINDEC-11- DZQKC	E	E1	03	PATCH

LOC	FROM	TO
2266	200	340
14146	200	340
16164	5227	4737
16166	Ø	160
160	-	5227
162	-	0
164	-	1375
166	-	207

History count started at REV. E

DOCUMENTATION AFFECTED
■ DIAGNOSTICS
□ TECH MANUAL
□ TESTER
□ TEST PROG
□ ENG SPEC
□ PURCH SPEC

FIELD SERVICE AFFECTED
■ YES □ NO
□ Customer Charge
□ Product Line Charge
□ PL.________
□ Contact Software Distribution Center for price.
□ Information Only

DISPOSITION CODES
00 – (DELETED)
01 – (DELETED)
02 – USE PRESENT STOCK UNTIL NEW STOCK AVAILABLE (PHASE IN)
03 – REWORK IMMEDIATELY (RETROFIT)
04 – (DELETED)
05 – (DELETED)
06 – DOCUMENT CORRECTION
07 – NEW ITEM (THIS ASSEMBLY)
08 – (DELETED)
09 – SCRAP IMMEDIATELY

APPROVAL SIGNATURES

	Typewritten	Hand Signature
DIAGNOSTIC ENGR.	Barry Irrgang	
MFG. ENGR.	B. Kellicker	
FIELD SERVICE	S. Johnson	
PRODUCT ENGR.	D. Testa	Testa

SEE REVERSE FOR INSTRUCTIONS (IN ORDER TO FILL OUT FORM PROPERLY)