DATA GENERAL
CORPORATION
Southboro,
Massachusetts 01772
(617) 485-9100

PROGRAM

Nova 800 Logic Test

TAPES

Binary: 095-000045-02

ABSTRACT

The Nova 800 Logic Test is a maintenance program designed to test the Nova 800 central processor unit. It is a gate by gate test of the logic used to implement the Nova 800 instruction set. The test does not include any input/output or optional equipment except multiply/divide.

097-000023-02

```
;                   NOVA-800 LOGIC TEST

;1. ABSTRACT
;         THE NOVA-800 LOGIC TEST IS A MAINTENANCE PROGRAM
;         DESIGNED TO TEST THE NOVA-800 CENTRAL PROCESSOR
;         UNIT.  IT IS A GATE BY GATE TEST OF THE LOGIC
;         USED TO IMPLEMENT THE NOVA-800 INSTRUCTION SET.
;         THE TEST DOES NOT INCLUDE ANY INPUT-OUTPUT OR
;         OPTIONAL EQUIPMENT EXCEPT MULTIPLY-DIVIDE.

;2. MACHINE REQUIREMENTS
;         NOVA-800 PROCESSOR
;         2K READ/WRITE MEMORY

;4. SWITCH SETTINGS
;         STARTING ADDRESS=000377, COMPLETE LOGIC TEST
;                          000374, MUL-DIV LOGIC TEST

;4. OPERATING PROCEDURE
;         LOAD THE PROGRAM VIA THE BINARY LOADER
;         SET THE SWITCHES TO 000377 OR 000374
;         PRESS START
;         MACHINE SHOULD HALT.  PRESS CONTINUE

;5. ERROR DESCRIPTION
;         THE HALT INSTRUCTION IS USED TO INDICATE ERRORS.
;         WHEN AN ERROR IS DETECTED RECORD THE STATE OF THE
;         MACHINE.  CONSULT THE LISTING FOR POSSIBLE CAUSES
;         OF FAILURE.  CONSTRUCT A LOOP WHICH WILL REPRODUCE
;         THE ERROR.  SCOPE THE LOGIC.

;6. PROGRAM DESCRIPTION
;         THIS PROGRAM IS A COLLECTION OF SMALL ROUTINES
;         EACH DESIGNED TO TEST A PORTION OF THE PROCESSOR
;         LOGIC.  EACH ROUTINE IS DESIGNED TO TEST AS SMALL
;         A PART OF THE LOGIC AS POSSIBLE.  EACH TEST IN THE
;         SEQUENCE IS BASED OF THE PREVIOUS TEST WORKING

;         THE GROUP OF TESTS AT THE END OF THE PROGRAM
;         FORM A CHECK OF THE MULTIPLY DIVIDE OPTION.
;         WHEN THE PROGRAM IS STARTED AT LOC 377 THESE
;         ADDITIONAL TESTS ARE PERFORMED IF THE PROGRAM
;         FINDS THAT THE MUL-DIV OPTION IS IN.  WHEN
;         THE PROGRAM IS STARTED AT LOC 374 ONLY THE
;         MUL-DIV TESTS ARE RUN.

;7. MISC
;         THE TIME FOR ONE COMPLETE PASS IS MEASURED
;         IN MILLISECONDS.
```

```
;8. TESTING PHILOSOPHY
;       THE NOVA-800 LOGIC TEST IS THE BASIC LOGIC TEST
;       FOR THIS MACHINE.  SINCE THE PROGRAM RUNS WITHIN
;       THE PROCESSOR IT IS TESTING (SELF TEST) OBVIOUSLY
;       IT MUST OPERATE UNDER SOME CONSTRAINTS.  A GUIDE
;       TO USING THIS PROGRAM WITHIN THE CONSTRAINTS IS
;       GIVEN BELOW

;8.1 PROGRAM LOADING
;       THE NOVA-800 LOGIC TEST IS DISTRIBUTED IS BINARY
;       FORMAT ON PAPER TAPE.  IF THE MACHINE TO BE TESTED
;       CAN LOAD THE PROGRAM VIA THE BINARY LOADER THE
;       COMPUTER IS IN PRETTY GOOD SHAPE TO BEGIN WITH.
;       THIS IS NOT TO SAY HOWEVER THAT THE LOGIC TEST WILL
;       NOT SUBSEQUENTLY FIND AN OBSCURE OR MARGINAL FAULT.
;       IDEALLY THE LOGIC TEST WILL BE LOADED INTO MEMORY
;       ON ANOTHER COMPUTER AND THE MEMORY THEN INSERTED INTO
;       THE NOVA-800 TO BE TESTED.  EVEN HERE A CERTAIN
;       AMOUNT OF LOGIC MUST BE FUNCTIONING PROPERLY
;       TO ALLOW SELF TEST.  THE MACHINE MUST BE ABLE TO
;       PERFORM A HALT INSTRUCTION WHICH IMPLIES THAT
;       BASIC TIMING, MEMORY CONTROL, INSTRUCTION REGISTER
;       ETC ARE IN FAIR CONDITION.
;       IF THE PROGRAM CAN BE LOADED AND STARTED, AND IF A
;       HALT CAN BE PERFORMED, THE PROGRAM THEN BECOMES
;       A USEFUL TOOL FOR FIXING MALFUNCTIONS.  IF NOT
;       ---- SEE BELOW.

;8.2 ALTERNATIVES
;       AS SOMETIMES HAPPENS THE COMPUTER DOES NOT WORK
;       WELL ENOUGH TO ALLOW SELF TESTING.  THUS
;       THE NOVA-800 LOGIC TEST WILL BE OF NO USE IN
;       FIXING THE PROBLEM.  OUTLINED HERE ARE SOME COMMOM
;       ALTERNATIVES TO THE LOGIC TEST PROGRAM.  SINCE
;       THE SYMPTOMS AND FAILURE CONDITTONS CAN BE SO VARIED
;       SPECIAL INGENUITY IS OFTEN REQUIRED TO FIND
;       THIS TYPE FAILURE.

;8.2.1 MEMORY FAILURES
;       IN MULTIPLE MEMORY SYSTEMS SWITCH CORE STACKS.
;       WITH THE FAILING STACK IN UPPER MEMORY A MEMORY
;       TEST PROGRAM CAN BE LOADED TO AID IN FINDING
;       THE TROUBLE.
;       IN 4K MEMORY SYSTEMS MOST HARD FAILURES CAN
;       BE FOUND USING THE EXAMINE AND DEPOSIT SWITCHES.
```

```
;8.2.2 PROCESSOR FAILURES
;       TEST ALL SWITCH FUNCTIONS: EXAMINE,DEPOSIT,LOAD AC
;       EXAMINE AC, ETC.  CONSOLE SWITCH COMMANDS EXERCISE
;       A GOOD PORTION OF THE PROCESSOR LOGIC.
;       KEY IN SHORT LOOPS AND RUN THEM: MEMORY STEP,
;       INSTRUCTION STEP, FULL SPEED.  I.E.

;       I       JMP .               PAGE 0
;                                   RELATIVE

;       II      HALT

;       III     MOV 0,0
;               HALT
;               HALT

;       IV      ETC...

;       IF ALL GOES WELL FINISH WITH A SHORT MEMORY ADDRESS
;       TEST.  SUCH AS -

;       0       LDA 2,12
;       1       STA 2,0,2
;       2       INC 2,2,SZR
;       3       JMP .-2
;       4       LDA 2,13
;       5       INC 2,2
;       6       LDA 0,0,2
;       7       SUB# 0,2,SNR
;       10      JMP .-3
;       11      HALT
;       12      14
;       13      13
```

```
        000000            .LOC 0
  00000 000000            0                  ;INTERRUPT ADDRESS
  00001 000000            0                  ;GO TO THIS ADDRESS
  00002 063077            HALT               ;LOOK AT TEST THAT
  00003 000002            2                  ;WAS INTERRUPTED FOR
                                             ;FAILURE SUGGESTIONS

        000045            .LOC 45
  00045 000050            EGGS

        000050            .LOC 50
  00050 000000 EGGS:      0                  ;HEN FLAG
  00051 000000            0                  ;DEVICE CODE
  00052 000000            0                  ;ASR37 FLAG
  00053 000000            0                  ;PASS COUNTER

  00054 000054            .
  00055 000000 K0:        0
  00056 000056            .
  00057 177777 KONES:     177777
  00060 000060            .
  00061 000001 KB15:      1
  00062 000062            .
  00063 100000 KB0:       100000
  00064 040000 KB1:       40000
  00065 020000 KB2:       20000
  00066 010000 KB3:       10000
  00067 004000 KB4:       4000
  00070 002000 KB5:       2000
  00071 001000 KB6:       1000
  00072 000400 KB7:       400
  00073 000200 KB8:       200
  00074 000100 KB9:       100
  00075 000040 KB10:      40
  00076 000020 KB11:      20
  00077 000010 KB12:      10
  00100 000004 KB13:      4
  00101 000002 KB14:      2

        000061 K1=KB15
  00102 000003 K3:        3
  00103 000007 K7:        7
  00104 000017 K17:       17
  00105 000037 K37:       37
  00106 000077 K77:       77
  00107 000177 K177:      177
  00110 000377 K377:      377
  00111 000777 K777:      777
  00112 001777 K1777:     1777
  00113 003777 K3777:     3777
  00114 007777 K7777:     7777
  00115 017777 K1.4K:     17777
  00116 037777 K3.4K:     37777
  00117 077777 K7.4K:     77777
  00120 177777 K17.4K:    177777
```

```
    ØØ121 Ø77777 KBØC:    77777
    ØØ122 137777 KB1C:    137777
    ØØ123 157777 KB2C:    157777
    ØØ124 167777 KB3C:    167777
    ØØ125 173777 KB4C:    173777
    ØØ126 175777 KB5C:    175777
    ØØ127 176777 KB6C:    176777
    ØØ13Ø 177377 KB7C:    177377
    ØØ131 177577 KB8C:    177577
    ØØ132 177677 KB9C:    177677
    ØØ133 177737 KB1ØC:   177737
    ØØ134 177757 KB11C:   177757
    ØØ135 177767 KB12C:   177767
    ØØ136 177773 KB13C:   177773
    ØØ137 177775 KB14C:   177775
    ØØ14Ø 177776 KB15C:   177776

    ØØ141 152525 KEB:     152525
    ØØ142 Ø25252 KOB:     Ø25252

    ØØ143 ØØØØØØ INDIR:   Ø
    ØØ144 ØØØØ61          KB15
    ØØ145 ØØØØØØ INDR1:   Ø
    ØØ146 1ØØ143          INDIR+1ØØØØØ

    ØØ147 177775 M3:      -3
    ØØ15Ø ØØØØØØ PGØ:     Ø

    ØØ151 14ØØØØ BØT1:    14ØØØØ
    ØØ152 16ØØØØ BØT2:    16ØØØØ
    ØØ153 17ØØØØ BØT3:    17ØØØØ
    ØØ154 174ØØØ BØT4:    174ØØØ
    ØØ155 176ØØØ BØT5:    176ØØØ
    ØØ156 177ØØØ BØT6:    177ØØØ
    ØØ157 1774ØØ BØT7:    1774ØØ
    ØØ16Ø 1776ØØ BØT8:    1776ØØ
    ØØ161 1777ØØ BØT9:    1777ØØ
    ØØ162 17774Ø BØT1Ø:   17774Ø
    ØØ163 17776Ø BØT11:   17776Ø
    ØØ164 17777Ø BØT12:   17777Ø
    ØØ165 177774 BØT13:   177774
    ØØ166 177776 BØT14:   177776
    ØØ167 177777 BØT15:   177777

    ØØ17Ø ØØØØØØ LADDR:   Ø
```

```
        000374          .LOC 374

  00374 002375          JMP @.+1
  00375 002442          MULDIV

        000376          .LOC 376

  00376 000402 LOOP:    JMP .+2

  00377 063077          HALT                ;PROCESSOR MUST BE ABLE TO
                                            ;TO EXECUTE A HALT INSTRUCTION
                                            ;IF THIS FAILS CONSULT
                                            ;WRITEUP "TESTING PHILOSOPHY"

  00400 101001 A00:     MOV 0,0,SKP         ;TEST PC SKIP.  CHECK
  00401 063077          HALT                ;INC PC, FETCH SKIP,ETC.

  00402 101000 A01:     MOV 0,0             ;CHECK FOR ILLEGAL SKIP
  00403 101001          MOV 0,0,SKP
  00404 063077          HALT

  00405 101043 A02:     MOVO 0,0,SNC        ;CHECK "CRY TEST",
  00406 063077          HALT                ;FETCH SKIP

  00407 101022 A03:     MOVZ 0,0,SZC        ;CHECK "CRY TEST",
  00410 063077          HALT                ;FETCH SKIP

  00411 101042 A04:     MOVO 0,0,SZC        ;CHECK "CRY TEST",
  00412 101001          MOV 0,0,SKP         ;FETCH SKIP
  00413 063077          HALT

  00414 101023 A05:     MOVZ 0,0,SNC        ;CHECK "CRY TEST",
  00415 101001          MOV 0,0,SKP         ;FETCH SKIP
  00416 063077          HALT

  00417 101040 A06:     MOVO 0,0            ;CHECK LOAD CRY
  00420 101003          MOV 0,0,SNC         ;CRY SET
  00421 063077          HALT

  00422 101020 A07:     MOVZ 0,0            ;CHECK LOAD CRY
  00423 101002          MOV 0,0,SZC         ;CRY SET
  00424 063077          HALT

  00425 101040 A08:     MOVO 0,0            ;CHECK CRY
  00426 101062          MOVC 0,0,SZC        ;COMPLIMENT
  00427 063077          HALT

  00430 101020 A09:     MOVZ 0,0            ;CHECK CRY
  00431 101063          MOVC 0,0,SNC        ;COMPLIMENT
  00432 063077          HALT

  00433 101040 A10:     MOVO 0,0            ;CHECK FOR NO
  00434 101033          MOVZ# 0,0,SNC       ;LOAD CRY
  00435 101003          MOV 0,0,SNC
  00436 063077          HALT
```

```
     00437 101020 A11:     MOVZ 0,0          ;CHECK FOR NO
     00440 101052          MOVO# 0,0,SZC     ;LOAD CRY
     00441 101002          MOV 0,0,SZC
     00442 063077          HALT

     00443 101040 A12:     MOVO 0,0          ;CHECK FOR NO
     00444 101073          MOVC# 0,0,SNC     ;LOAD CRY
     00445 101062          MOVC 0,0,SZC
     00446 063077          HALT

     00447 101020 A13:     MOVZ 0,0          ;CHECK FOR NO
     00450 101072          MOVC# 0,0,SZC     ;LOAD CRY
     00451 101063          MOVC 0,0,SNC
     00452 063077          HALT

     00453 020055 A14:     LDA 0,K0          ;FIRST LDA, CHECK
     00454 101004          MOV 0,0,SZR       ;ADDER=0,FETCH SKIP
     00455 063077          HALT              ;MEM-AR DATA PATH
                                             ;THRU ADDER

     00456 020055 A15:     LDA 0,K0          ;CHECK ADDER=0
     00457 101005          MOV 0,0,SNR       ;NO FETCH SKIP
     00460 101004          MOV 0,0,SZR
     00461 063077          HALT

     00462 020063 A16:     LDA 0,KB0         ;CHECK MEM-AR-AC0
     00463 101004          MOV 0,0,SZR       ;DATA PATH
     00464 101005          MOV 0,0,SNR       ;FOR BIT 0=1
     00465 063077          HALT

     00466 020064 A17:     LDA 0,KB1         ;BIT 1
     00467 101004          MOV 0,0,SZR
     00470 101005          MOV 0,0,SNR
     00471 063077          HALT

     00472 020065 A18:     LDA 0,KB2         ;BIT 2
     00473 101004          MOV 0,0,SZR
     00474 101005          MOV 0,0,SNR
     00475 063077          HALT

     00476 020066 A19:     LDA 0,KB3         ;BIT 3
     00477 101004          MOV 0,0,SZR
     00500 101005          MOV 0,0,SNR
     00501 063077          HALT

     00502 020067 A20:     LDA 0,KB4         ;BIT 4
     00503 101004          MOV 0,0,SZR
     00504 101005          MOV 0,0,SNR
     00505 063077          HALT

     00506 020070 A21:     LDA 0,KB5         ;BIT 5
     00507 101004          MOV 0,0,SZR
     00510 101005          MOV 0,0,SNR
     00511 063077          HALT
```

```
A ØØØ8  .MAIN

    ØØ512 Ø2ØØ71 A22:    LDA Ø,KB6        ;BIT 6
    ØØ513 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ514 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ515 Ø63Ø77         HALT

    ØØ516 Ø2ØØ72 A23:    LDA Ø,KB7        ;BIT 7
    ØØ517 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ52Ø 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ521 Ø63Ø77         HALT

    ØØ522 Ø2ØØ73 A24:    LDA Ø,KB8        ;BIT 8
    ØØ523 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ524 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ525 Ø63Ø77         HALT

    ØØ526 Ø2ØØ74 A25:    LDA Ø,KB9        ;BIT 9
    ØØ527 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ53Ø 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ531 Ø63Ø77         HALT

    ØØ532 Ø2ØØ75 A26:    LDA Ø,KB1Ø       ;BIT 1Ø
    ØØ533 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ534 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ535 Ø63Ø77         HALT

    ØØ536 Ø2ØØ76 A27:    LDA Ø,KB11       ;BIT 11
    ØØ537 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ54Ø 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ541 Ø63Ø77         HALT

    ØØ542 Ø2ØØ77 A28:    LDA Ø,KB12       ;BIT 12
    ØØ543 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ544 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ545 Ø63Ø77         HALT

    ØØ546 Ø2Ø1ØØ A29:    LDA Ø,KB13       ;BIT 13
    ØØ547 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ55Ø 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ551 Ø63Ø77         HALT

    ØØ552 Ø2Ø1Ø1 A3Ø:    LDA Ø,KB14       ;BIT 14
    ØØ553 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ554 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ555 Ø63Ø77         HALT

    ØØ556 Ø2ØØ61 A31:    LDA Ø,KB15       ;BIT 15
    ØØ557 1Ø1ØØ4         MOV Ø,Ø,SZR
    ØØ56Ø 1Ø1ØØ5         MOV Ø,Ø,SNR
    ØØ561 Ø63Ø77         HALT

    ØØ562 Ø2ØØ55 A32:    LDA Ø,KØ         ;CHECK AC SELECTION
    ØØ563 Ø24Ø64         LDA 1,KB1        ;AC-H-WRITE
    ØØ564 Ø3ØØ65         LDA 2,KB2        ;AC-L-WRITE
    ØØ565 Ø34Ø66         LDA 3,KB3        ;FOUR COMBINATIONS
    ØØ566 1Ø1ØØ4         MOV Ø,Ø,SZR      ;READ, SX-H-READ=Ø
    ØØ567 Ø63Ø77         HALT             ;      SX-L-READ=Ø
```

```
A ØØØ9  .MAIN

    ØØ57Ø Ø24Ø55 A33:      LDA 1,KØ         ;CHECK AC SELECTION
    ØØ571 Ø3ØØ64           LDA 2,KB1        ;AC-H-WRITE
    ØØ572 Ø34Ø65           LDA 3,KB2        ;AC-L-WRITE
    ØØ573 Ø2ØØ66           LDA Ø,KB3        ;FOUR COMBINATIONS
    ØØ574 125ØØ4           MOV 1,1,SZR      ;READ, SX-H-READ=Ø
    ØØ575 Ø63Ø77           HALT             ;       SX-L-READ=1

    ØØ576 Ø3ØØ55 A34:      LDA 2,KØ         ;CHECK AC SELECTION
    ØØ577 Ø34Ø64           LDA 3,KB1        ;AC-H-WRITE
    ØØ6ØØ Ø2ØØ65           LDA Ø,KB2        ;AC-L-WRITE
    ØØ6Ø1 Ø24Ø66           LDA 1,KB3        ;FOUR COMBNATIONS
    ØØ6Ø2 151ØØ4           MOV 2,2,SZR      ;READ, SX-H-READ=1
    ØØ6Ø3 Ø63Ø77           HALT             ;       SX-L-READ=Ø

    ØØ6Ø4 Ø34Ø55 A35:      LDA 3,KØ         ;CJECK AC SELECTION
    ØØ6Ø5 Ø2ØØ64           LDA Ø,KB1        ;AC-H-WRITE
    ØØ6Ø6 Ø24Ø65           LDA 1,KB2        ;AC-L-WRITE
    ØØ6Ø7 Ø3ØØ66           LDA 2,KB3        ;FOUR COMBINATIONS
    ØØ61Ø 175ØØ4           MOV 3,3,SZR      ;READ, SX-H-READ=1
    ØØ611 Ø63Ø77           HALT             ;       SX-L-READ=1

    ØØ612 1Ø1Ø2Ø A36:      MOVZ Ø,Ø         ;TEST CRY, SEL-L
    ØØ613 Ø2ØØ63           LDA Ø,KBØ        ;LEFT SHIFT
    ØØ614 1Ø11Ø3           MOVL Ø,Ø,SNC
    ØØ615 Ø63Ø77           HALT

    ØØ616 1Ø1Ø2Ø A37:      MOVZ Ø,Ø         ;TEST CRY, SEL-R
    ØØ617 Ø2ØØ61           LDA Ø,KB15       ;RIGHT SHIFT
    ØØ62Ø 1Ø12Ø3           MOVR Ø,Ø,SNC
    ØØ621 Ø63Ø77           HALT

    ØØ622 1Ø1343 A38:      MOVOS Ø,Ø,SNC    ;TEST CRY, SEL-S
    ØØ623 Ø63Ø77           HALT

    ØØ624 1Ø1322 A39:      MOVZS Ø,Ø,SZC    ;TEST CRY, SEL-S
    ØØ625 Ø63Ø77           HALT             ;SWAP

    ØØ626 Ø2ØØ61 A4Ø:      LDA Ø,KB15       ;ANOTHER CHECK ON
    ØØ627 1Ø1Ø27           MOVZ Ø,Ø,SBN     ;FETCH SKIP, INC PC
    ØØ63Ø 1Ø1ØØ6           MOV Ø,Ø,SEZ
    ØØ631 Ø63Ø77           HALT

    ØØ632 Ø2ØØ55 A41:      LDA Ø,KØ         ;AGAIN
    ØØ633 1Ø1Ø47           MOVO Ø,Ø,SBN
    ØØ634 1Ø1ØØ6           MOV Ø,Ø,SEZ
    ØØ635 Ø63Ø77           HALT

    ØØ636 Ø2ØØ55 A42:      LDA Ø,KØ         ;AGAIN
    ØØ637 1Ø1Ø26           MOVZ Ø,Ø,SEZ
    ØØ64Ø Ø63Ø77           HALT

    ØØ641 Ø2ØØ61 A43:      LDA Ø,KB15       ;AGAIN
    ØØ642 1Ø1Ø47           MOVO Ø,Ø,SBN
    ØØ643 Ø63Ø77           HALT

                           .EOT
```

```
0010  .MAIN

 00644 020057 B01:      LDA 0,KONES         ;CHECK SX-COM
 00645 104004           COM 0,1,SZR         ;INPUT TO SX-MULT
 00646 063077           HALT

 00647 020055 B02:      LDA 0,K0            ;SAME
 00650 104004           COM 0,1,SZR
 00651 130004           COM 1,2,SZR
 00652 063077           HALT

 00653 020063 B03:      LDA 0,KB0           ;TEST ADDER 0,
 00654 024121           LDA 1,KB0C          ;NO CARRIES
 00655 107023           ADDZ 0,1,SNC
 00656 130004           COM 1,2,SZR         ;100000 + 077777
 00657 063077           HALT

 00660 020064 B04:      LDA 0,KB1           ;TEST ADDER 1
 00661 024122           LDA 1,KB1C          ;NO CARRIES
 00662 107023           ADDZ 0,1,SNC
 00663 130004           COM 1,2,SZR         ;040000 + 137777
 00664 063077           HALT

 00665 020065 B05:      LDA 0,KB2           ;TEST ADDER 2
 00666 024123           LDA 1,KB2C          ;NO CARRIES
 00667 107023           ADDZ 0,1,SNC
 00670 130004           COM 1,2,SZR         ;020000 + 157777
 00671 063077           HALT

 00672 020066 B06:      LDA 0,KB3           ;TEST ADDER 3
 00673 024124           LDA 1,KB3C          ;NO CARRIES
 00674 107023           ADDZ 0,1,SNC
 00675 130004           COM 1,2,SZR         ;010000 + 167777
 00676 063077           HALT

 00677 020067 B07:      LDA 0,KB4           ;TEST ADDER 4
 00700 024125           LDA 1,KB4C          ;NO CARRIES
 00701 107042           ADDO 0,1,SZC
 00702 130004           COM 1,2,SZR         ;004000 + 173777
 00703 063077           HALT

 00704 020070 B08:      LDA 0,KB5           ;TEST ADDER 5
 00705 024126           LDA 1,KB5C          ;NO CARRIES
 00706 107023           ADDZ 0,1,SNC
 00707 130004           COM 1,2,SZR         ;002000 + 175777
 00710 063077           HALT

 00711 020071 B09:      LDA 0,KB6           ;TEST ADDER 6
 00712 024127           LDA 1,KB6C          ;NO CARRIES
 00713 107023           ADDZ 0,1,SNC
 00714 130004           COM 1,2,SZR         ;001000 + 176777
 00715 063077           HALT

 00716 020072 B10:      LDA 0,KB7           ;TEST ADDER 7
 00717 024130           LDA 1,KB7C          ;NO CARRIES
 00720 107023           ADDZ 0,1,SNC
 00721 130004           COM 1,2,SZR         ;000400 + 177377
 00722 063077           HALT
```

```
    00723 020073 B11:     LDA 0,KB8         ;TEST ADDER 8
    00724 024131          LDA 1,KB8C        ;NO CARRIES
    00725 107042          ADDO 0,1,SZC
    00726 130004          COM 1,2,SZR       ;000200 + 177577
    00727 063077          HALT

    00730 020074 B12:     LDA 0,KB9         ;TEST ADDER 9
    00731 024132          LDA 1,KB9C        ;NO CARRIES
    00732 107042          ADDO 0,1,SZC
    00733 130004          COM 1,2,SZR       ;000100 + 177677
    00734 063077          HALT

    00735 020075 B13:     LDA 0,KB10        ;TEST ADDER 10
    00736 024133          LDA 1,KB10C       ;NO CARRIES
    00737 107042          ADDO 0,1,SZC
    00740 130004          COM 1,2,SZR       ;000040 + 177737
    00741 063077          HALT

    00742 020076 B14:     LDA 0,KB11        ;TEST ADDER 11
    00743 024134          LDA 1,KB11C       ;NO CARRIES
    00744 107023          ADDZ 0,1,SNC
    00745 130004          COM 1,2,SZR       ;000020 + 177757
    00746 063077          HALT

    00747 020077 B15:     LDA 0,KB12        ;TEST ADDER 12
    00750 024135          LDA 1,KB12C       ;NO CARRIES
    00751 107023          ADDZ 0,1,SNC
    00752 130004          COM 1,2,SZR       ;000010 + 177767
    00753 063077          HALT

    00754 020100 B16:     LDA 0,KB13        ;TEST ADDER 13
    00755 024136          LDA 1,KB13C       ;NO CARRIES
    00756 107023          ADDZ 0,1,SNC
    00757 130004          COM 1,2,SZR       ;000004 + 177773
    00760 063077          HALT

    00761 020101 B17:     LDA 0,KB14        ;TEST ADDER 14
    00762 024137          LDA 1,KB14C       ;NO CARRIES
    00763 107023          ADDZ 0,1,SNC
    00764 130004          COM 1,2,SZR       ;000002 + 177775
    00765 063077          HALT

    00766 020061 B18:     LDA 0,KB15        ;TEST ADDER 15
    00767 024140          LDA 1,KB15C       ;NO CARRIES
    00770 107042          ADDO 0,1,SZC
    00771 130004          COM 1,2,SZR       ;000001 + 177776
    00772 063077          HALT

    00773 020061 B19:     LDA 0,KB15        ;CHECK ACD(0) READ
    00774 024101          LDA 1,KB14        ;SELECTION
    00775 030100          LDA 2,KB13
    00776 034077          LDA 3,KB12        ;D-H-READ=0
    00777 102023          ADCZ 0,0,SNC      ;D-L-READ=0
    01000 100004          COM 0,0,SZR
    01001 063077          HALT
    01002 020061 B20:     LDA 0,KB15          ;CHECK ACD(1) READ
    01003 024101          LDA 1,KB14          ;SELECTION
    01004 030100          LDA 2,KB13
    01005 034077          LDA 3,KB12          ;D=H=READ=0
    01006 126023          ADCZ 1,1,SNC        ;D=L=READ=1
    01007 124004          COM 1,1,SZR
    01010 063077          HALT

    01011 020061 B21:     LDA 0,KB15          ;CHECK ACD(2) READ
    01012 024101          LDA 1,KB14          ;SELECTION
    01013 030100          LDA 2,KB13
    01014 034077          LDA 3,KB12          ;D=H=READ=1
    01015 152023          ADCZ 2,2,SNC        ;D=L=READ=0
    01016 150004          COM 2,2,SZR
    01017 063077          HALT

    01020 020061 B22:     LDA 0,KB15          ;CHECK ACD(3) READ
    01021 024101          LDA 1,KB14          ;SELECTION
    01022 030100          LDA 2,KB13
    01023 034077          LDA 3,KB12          ;D=H=READ=1
    01024 176023          ADCZ 3,3,SNC        ;D=L=READ=1
    01025 174004          COM 3,3,SZR
    01026 063077          HALT

    01027 020055 B23:     LDA 0,K0            ;SHIFT RIGHT, CRY=BIT0
    01030 030121          LDA 2,KB0C
    01031 105240          MOVOR 0,1           ;SEL=R, SUM CRY
    01032 133003          ADD 1,2,SNC         ;GATE ON SUM 0
    01033 154004          COM 2,3,SZR
    01034 063077          HALT

    01035 020063 B24:     LDA 0,KB0           ;SHIFT RIGHT, BIT 0=1
    01036 030122          LDA 2,KB1C
    01037 105220          MOVZR 0,1           ;SEL=R, ADDER 0
    01040 133003          ADD 1,2,SNC         ;GATE ON SUM 1
    01041 154004          COM 2,3,SZR
    01042 063077          HALT

    01043 020064 B25:     LDA 0,KB1           ;SHIFT RIGHT, BIT 1 TO 2
    01044 030123          LDA 2,KB2C
    01045 105220          MOVZR 0,1           ;SEL=R, ADDER 1
    01046 133003          ADD 1,2,SNC         ;GATE ON SUM 2
    01047 154004          COM 2,3,SZR
    01050 063077          HALT

    01051 020065 B26:     LDA 0,KB2           ;SHIFT RIGHT, BIT 2 TO 3
    01052 030124          LDA 2,KB3C
    01053 105220          MOVZR 0,1           ;SEL=R, ADDER 2
    01054 133003          ADD 1,2,SNC         ;GATE ON SUM 3
    01055 154004          COM 2,3,SZR
    01056 063077          HALT

    01057 020066 B27:     LDA 0,KB3           ;SHIFT RIGHT, BIT 3 TO 4
    01060 030125          LDA 2,KB4C
    01061 105220          MOVZR 0,1           ;SEL=R, ADDER 3
    01062 133003          ADD 1,2,SNC         ;GATE ON SUM 4
    01063 154004          COM 2,3,SZR
    01064 063077          HALT
```

```
     01065 020067 B28:      LDA 0,KB4         ;SHIFT RIGHT, BIT 4 TO 5
     01066 030126           LDA 2,KB5C
     01067 105220           MOVZR 0,1         ;SEL=R, ADDER 4
     01070 133003           ADD 1,2,SNC       ;GATE ON SUM 5
     01071 154004           COM 2,3,SZR
     01072 063077           HALT

     01073 020070 B29:      LDA 0,KB5         ;SHIFT RIGHT, BIT 5 TO 6
     01074 030127           LDA 2,KB6C
     01075 105220           MOVZR 0,1         ;SEL=R ADDER 5
     01076 133003           ADD 1,2,SNC       ;GATE ON SUM 6
     01077 154004           COM 2,3,SZR
     01100 063077           HALT

     01101 020071 B30:      LDA 0,KB6         ;SHIFT RIGHT, BIT 6 TO 7
     01102 030130           LDA 2,KB7C
     01103 105220           MOVZR 0,1         ;SEL=R, ADDER 6
     01104 133003           ADD 1,2,SNC       ;GATE ON SUM 7
     01105 154004           COM 2,3,SZR
     01106 063077           HALT

     01107 020072 B31:      LDA 0,KB7         ;SHIFT RIGHT, BIT 7 TO 8
     01110 030131           LDA 2,KB8C
     01111 105220           MOVZR 0,1         ;SEL=R, ADDER 7
     01112 133003           ADD 1,2,SNC       ;GATE ON SUM 8
     01113 154004           COM 2,3,SZR
     01114 063077           HALT

     01115 020073 B32:      LDA 0,KB8         ;SHIFT RIGHT, BIT 8 TO 9
     01116 030132           LDA 2,KB9C
     01117 105220           MOVZR 0,1         ;SEL=R, ADDER 8
     01120 133003           ADD 1,2,SNC       ;GATE ON SUM 9
     01121 154004           COM 2,3,SZR
     01122 063077           HALT

     01123 020074 B33:      LDA 0,KB9         ;SHIFT RIGHT, BIT 9 TO 10
     01124 030133           LDA 2,KB10C
     01125 105220           MOVZR 0,1         ;SEL=R, ADDER 9
     01126 133003           ADD 1,2,SNC       ;GATE ON SUM 10
     01127 154004           COM 2,3,SZR
     01130 063077           HALT

     01131 020075 B34:      LDA 0,KB10        ;SHIFT RIGHT, BIT 10 TO 11
     01132 030134           LDA 2,KB11C
     01133 105220           MOVZR 0,1         ;SEL=R, ADDER 10
     01134 133003           ADD 1,2,SNC       ;GATE ON SUM 11
     01135 154004           COM 2,3,SZR
     01136 063077           HALT

     01137 020076 B35:      LDA 0,KB11        ;SHIFT RIGHT, BIT 11 TO 12
     01140 030135           LDA 2,KB12C
     01141 105220           MOVZR 0,1         ;SEL=R, ADDER 11
     01142 133003           ADD 1,2,SNC       ;GATE ON SUM 12
     01143 154004           COM 2,3,SZR
     01144 063077           HALT
```

```
01145 020077 B36:       LDA 0,KB12         ;SHIFT RIGHT, BIT 12 TO 13
01146 030136            LDA 2,KB13C
01147 105220            MOVZR 0,1          ;SEL=R, ADDER 12
01150 133003            ADD 1,2,SNC        ;GATE ON SUM 13
01151 154004            COM 2,3,SZR
01152 063077            HALT

01153 020100 B37:       LDA 0,KB13         ;SHIFT RIGHT, BIT 13 TO 14
01154 030137            LDA 2,KB14C
01155 105220            MOVZR 0,1          ;SEL=R, ADDER 13
01156 133003            ADD 1,2,SNC        ;GATE ON SUM 14
01157 154004            COM 2,3,SZR
01160 063077            HALT

01161 020101 B38:       LDA 0,KB14         ;SHIFT RIGHT, BIT 14 TO 15
01162 030140            LDA 2,KB15C
01163 105220            MOVZR 0,1          ;SEL=R ADDER 14
01164 133003            ADD 1,2,SNC        ;GATE ON SUM 15
01165 154004            COM 2,3,SZR
01166 063077            HALT

01167 020055 B39:       LDA 0,K0           ;SHIFT LEFT, CRY TO BIT 15
01170 030140            LDA 2,KB15C
01171 105140            MOVOL 0,1          ;SEL=L, SUM CRY
01172 133003            ADD 1,2,SNC        ;GATE ON SUM 15
01173 154004            COM 2,3,SZR
01174 063077            HALT

01175 020061 B40:       LDA 0,KB15         ;SHIFT LEFT, BIT 15 TO 14
01176 030137            LDA 2,KB14C
01177 105120            MOVZL 0,1          ;SEL=L, ADDER 15
01200 133003            ADD 1,2,SNC        ;GATE ON SUM 14
01201 154004            COM 2,3,SZR
01202 063077            HALT

01203 020101 B41:       LDA 0,KB14         ;SHIFT LEFT, BIT 14 TO 13
01204 030136            LDA 2,KB13C
01205 105120            MOVZL 0,1          ;SEL=L, ADDER 14
01206 133003            ADD 1,2,SNC        ;GATE ON SUM 13
01207 154004            COM 2,3,SZR
01210 063077            HALT

01211 020100            LDA 0,KB13         ;SHIFT LEFT, BIT 13 TO 12
01212 030135            LDA 2,KB12C
01213 105120            MOVZL 0,1          ;SEL=L, ADDER 13
01214 133003            ADD 1,2,SNC        ;GATE ON SUM12
01215 154004            COM 2,3,SZR
01216 063077            HALT

01217 020077 B43:       LDA 0,KB12         ;SHIFT LEFT, BIT 12 TO 11
01220 030134            LDA 2,KB11C
01221 105120            MOVZL 0,1          ;SEL=L, ADDER 12
01222 133003            ADD 1,2,SNC        ;GATE ON SUM 11
01223 154004            COM 2,3,SZR
01224 063077            HALT
```

```
01225 020076 B44:      LDA 0,KB11        ;SHIFT LEFT, BIT 11 TO 10
01226 030133           LDA 2,KB10C
01227 105120           MOVZL 0,1         ;SEL=L, ADDER 11
01230 133003           ADD 1,2,SNC       ;GATE ON SUM 10
01231 154004           COM 2,3,SZR
01232 063077           HALT

01233 020075 B45:      LDA 0,KB10        ;SHIFT LEFT, BIT 10 TO 9
01234 030132           LDA 2,KB9C
01235 105120           MOVZL 0,1         ;SEL=L, ADDER 10
01236 133003           ADD 1,2,SNC       ;GATE ON SUM 9
01237 154004           COM 2,3,SZR
01240 063077           HALT

01241 020074 B46:      LDA 0,KB9         ;SHIFT LEFT, BIT 9 TO 8
01242 030131           LDA 2 KB8C
01243 105120           MOVZL 0,1         ;SE;=L, ADDER 9
01244 133003           ADD 1,2,SNC       ;GATE ON SUM 8
01245 154004           COM 2,3,SZR
01246 063077           HALT

01247 020073 B47:      LDA 0,KB8         ;SHIFT LEFT, BIT 8 TO 7
01250 030130           LDA 2,KB7C
01251 105120           MOVZL 0,1         ;SEL=L, ADDER 8
01252 133003           ADD 1,2,SNC       ;GATE ON SUM 7
01253 154004           COM 2,3,SZR
01254 063077           HALT

01255 020072 B48:      LDA 0,KB7         ;SHIFT LEFT, BIT 7 TO 6
01256 030127           LDA 2,KB6C
01257 105120           MOVZL 0,1         ;SEL=L, ADDER 7
01260 133003           ADD 1,2,SNC       ;GATE ON SUM 6
01261 154004           COM 2,3,SZR
01262 063077           HALT

01263 020071 B49:      LDA 0,KB6         ;SHIFT LEFT, BIT 6 TO 5
01264 030126           LDA 2,KB5C
01265 105120           MOVZL 0,1         ;SEL=L, ADDER 6
01266 133003           ADD 1,2,SNC       ;GATE ON SUM 5
01267 154004           COM 2,3,SZR
01270 063077           HALT

01271 020070 B50:      LDA 0,KB5         ;SHIFT LEFT, BIT 5 TO 4
01272 030125           LDA 2,KB4C
01273 105120           MOVZL 0,1         ;SEL=L, ADDER 5
01274 133003           ADD 1,2,SNC       ;GATE ON SUM 4
01275 154004           COM 2,3,SZR
01276 063077           HALT

01277 020067 B51:      LDA 0,KB4         ;SHIFT LEFT, BIT 4 TO 3
01300 030124           LDA 2,KB3C
01301 105120           MOVZL 0,1         ;SEL=L, ADDER 4
01302 133003           ADD 1,2,SNC       ;GATE ON SUM 3
01303 154004           COM 2,3,SZR
01304 063077           HALT
```

```
    01305 020066 B52:     LDA 0,KB3          ;SHIFT LEFT, BIT 3 TO 2
    01306 030123          LDA 2,KB2C
    01307 105120          MOVZL 0,1          ;SEL=L, ADDER 3
    01310 133003          ADD 1,2,SNC        ;GATE ON SUM 2
    01311 154004          COM 2,3,SZR
    01312 063077          HALT

    01313 020065 B53:     LDA 0,KB2          ;SHIFT LEFT, BIT 2 TO 1
    01314 030122          LDA 2,KB1C
    01315 105120          MOVZL 0,1          ;SEL=L, ADDER 2
    01316 133003          ADD 1,2,SNC        ;GATE ON SUM 1
    01317 154004          COM 2,3,SZR
    01320 063077          HALT

    01321 020064 B54:     LDA 0,KB1          ;SHIFT LEFT, BIT 1 TO 0
    01322 030121          LDA 2,KB0C
    01323 105120          MOVZL 0,1          ;SEL=L, ADDER 1
    01324 133003          ADD 1,2,SNC        ;GATE ON SUM 0
    01325 154004          COM 2,3,SZR
    01326 063077          HALT

    01327 020061 B55:     LDA 0,KB15         ;SWAP, BIT 15 TO 7
    01330 030130          LDA 2,KB7C
    01331 105300          MOVS 0,1           ;SEL=S, ADDER 15
    01332 133000          ADD 1,2            ;GATE ON SUM 7
    01333 154004          COM 2,3,SZR
    01334 063077          HALT

    01335 020101 B56:     LDA 0,KB14         ;SWAP, BIT 14 TO 6
    01336 030127          LDA 2,KB6C
    01337 105300          MOVS 0,1           ;SEL=S, ADDER 14
    01340 133000          ADD 1,2            ;GATE ON SUM 6
    01341 154004          COM 2,3,SZR
    01342 063077          HALT

    01343 020100 B57:     LDA 0,KB13         ;SWAP, BIT 13 TO 5
    01344 030126          LDA 2,KB5C
    01345 105300          MOVS 0,1           ;SEL=S, ADDER 13
    01346 133000          ADD 1,2            ;GATE ON SUM 5
    01347 154004          COM 2,3,SZR
    01350 063077          HALT

    01351 020077 B58:     LDA 0,KB12         ;SWAP BIT 12 TO 4
    01352 030125          LDA 2,KB4C
    01353 105300          MOVS 0,1           ;SEL=S, ADDER 12
    01354 133000          ADD 1,2            ;GATE ON SUM 4
    01355 154004          COM 2,3,SZR
    01356 063077          HALT

    01357 020076 B59:     LDA 0,KB11         ;SWAP BIT 11 TO 3
    01360 030124          LDA 2,KB3C
    01361 105300          MOVS 0,1           ;SEL=S, ADDER 11
    01362 133000          ADD 1,2            ;GATE ON SUM 3
    01363 154004          COM 2,3,SZR
    01364 063077          HALT
```

```
     01365 020075 B60:     LDA 0,KB10          ;SWAP, BIT 10 TO 2
     01366 030123          LDA 2,KB2C
     01367 105300          MOVS 0,1            ;SEL=S, ADDER 10
     01370 133000          ADD 1,2             ;GATE ON SUM 2
     01371 154004          COM 2,3,SZR
     01372 063077          HALT

     01373 020074 B61:     LDA 0,KB9           ;SWAP BIT 9 TO 1
     01374 030122          LDA 2,KB1C
     01375 105300          MOVS 0,1            ;SEL=S, ADDER 9
     01376 133000          ADD 1,2             ;GATE ON SUM 1
     01377 154004          COM 2,3,SZR
     01400 063077          HALT

     01401 020073 B62:     LDA 0,KB8           ;SWAP BIT 8 TO 0
     01402 030121          LDA 2,KB0C
     01403 105300          MOVS 0,1            ;SEL=S, ADDER 8
     01404 133000          ADD 1,2             ;GATE ON SUM 0
     01405 154004          COM 2,3,SZR
     01406 063077          HALT

     01407 020072 B63:     LDA 0,KB7           ;SWAP BIT 7 TO 15
     01410 030140          LDA 2,KB15C
     01411 105300          MOVS 0,1            ;SEL=S, ADDER 7
     01412 133000          ADD 1,2             ;GATE ON SUM 15
     01413 154004          COM 2,3,SZR
     01414 063077          HALT

     01415 020071 B64:     LDA 0,KB6           ;SWAP BIT 6 TO 14
     01416 030137          LDA 2,KB14C
     01417 105300          MOVS 0,1            ;SEL=S, ADDER 6
     01420 133000          ADD 1,2             ;GATE ON SUM 14
     01421 154004          COM 2,3,SZR
     01422 063077          HALT

     01423 020070 B65:     LDA 0,KB5           ;SWAP BIT 5 TO 13
     01424 030136          LDA 2,KB13C
     01425 105300          MOVS 0,1            ;SEL=S, ADDER 5
     01426 133000          ADD 1,2             ;GATE ON SUM 13
     01427 154004          COM 2,3,SZR
     01430 063077          HALT

     01431 020067 B66:     LDA 0,KB4           ;SWAP BIT 4 TO 12
     01432 030135          LDA 2,KB12C
     01433 105300          MOVS 0,1            ;SEL=S, ADDER 4
     01434 133000          ADD 1,2             ;GATE ON SUM 12
     01435 154004          COM 2,3,SZR
     01436 063077          HALT

     01437 020066 B67:     LDA 0,KB3           ;SWAP BIT 3 TO 11
     01440 030134          LDA 2,KB11C
     01441 105300          MOVS 0,1            ;SEL=S, ADDER 3
     01442 133000          ADD 1,2             ;GATE ON SUM 11
     01443 154004          COM 2,3,SZR
     01444 063077          HALT
```

```
  01445 020065 B68:    LDA 0,KB2        ;SWAP BIT 2 TO 10
  01446 030133         LDA 2,KB10C
  01447 105300         MOVS 0,1         ;SEL=S, ADDER 2
  01450 133000         ADD 1,2          ;GATE ON SUM 10
  01451 154004         COM 2,3,SZR
  01452 063077         HALT

  01453 020064 B69:    LDA 0,KB1        ;SWAP BIT 1 TO 9
  01454 030132         LDA 2,KB9C
  01455 105300         MOVS 0,1         ;SEL=S, ADDER 1
  01456 133000         ADD 1,2          ;GATE ON SUM 9
  01457 154004         COM 2,3,SZR
  01460 063077         HALT

  01461 020063 B70:    LDA 0,KB0        ;SWAP BIT 0 YO 8
  01462 030131         LDA 2,KB8C
  01463 105300         MOVS 0,1         ;SEL=S, ADDER 0
  01464 133000         ADD 1,2          ;GATE ON SUM 8
  01465 154004         COM 2,3,SZR
  01466 063077         HALT

  01467 020063 C01:    LDA 0,KB0        ;TEST CARRY PROPAGATION
  01470 024063         LDA 1,KB0        ;CHECK ADDER 0 CARRY OUT
  01471 123022         ADDZ 1,0,SZC     ;100000 + 100000
  01472 101004         MOV 0,0,SZR      ;CR&CG MUST BOTH =1
  01473 063077         HALT             ;INPUT TO "CRY TO AR"

  01474 020151 C02:    LDA 0,B0T1       ;TEST CARRY PROPAGATION
  01475 024064         LDA 1,KB1        ;CHECK ADDER 1
  01476 123022         ADDZ 1,0,SZC     ;040000 + 140000
  01477 101004         MOV 0,0,SZR      ;CARRY FROM ADDER 1 TO 0
  01500 063077         HALT             ;IS PROBABLY BAD

  01501 020152 C03:    LDA 0,B0T2       ;TEST CARRY PROPAGATION
  01502 024065         LDA 1,KB2        ;ADDER 2 - LEFT
  01503 123022         ADDZ 1,0,SZC     ;020000 + 160000
  01504 101004         MOV 0,0,SZR      ;CARRY FROM ADDER 2 TO 1
  01505 063077         HALT             ;IS PROBABLY BAD
  01506 020153 C04:    LDA 0,B0T3       ;TEST CARRY PROPAGATION
  01507 024066         LDA 1,KB3        ;ADDER 3 - LEFT
  01510 123022         ADDZ 1,0,SZC     ;010000 + 170000
  01511 101004         MOV 0,0,SZR      ;CARRY FROM ADDER 3 TO 2
  01512 063077         HALT             ;IS PROBABLY BAD


  01513 020154 C05:    LDA 0,B0T4       ;TEST CARRY PROPAGATION
  01514 024067         LDA 1,KB4        ;ADDER 4 - LEFT
  01515 123022         ADDZ 1,0,SZC     ;004000 + 174000
  01516 101004         MOV 0,0,SZR      ;CHECK ADDER 4 CARRY OUT
  01517 063077         HALT             ;(PINS 14-15) TO ADDER
                                        ;3 CARRY IN (PINS 3-4)
                                        ;BOTH MUST BE HIGH!!

  01520 020155 C06:    LDA 0,B0T5       ;TEST CARRY PROPAGATION
  01521 024070         LDA 1,KB5        ;ADDER 5 - LEFT
  01522 123022         ADDZ 1,0,SZC     ;002000 + 176000
  01523 101004         MOV 0,0,SZR      ;CARRY FROM ADDER 5 TO 4
  01524 063077         HALT             ;IS PROBABLY BAD
```

```
01525 020156 C07:     LDA 0,B0T6          ;TEST CARRY PROPAGATION
01526 024071          LDA 1,KB6           ;ADDER 6 - LEFT
01527 123022          ADDZ 1,0,SZC        ;001000 + 177000
01530 101004          MOV 0,0,SZR         ;CARRY FROM ADDER 6 TO 5
01531 063077          HALT                ;IS PROBABLY BAD

01532 020157 C08:     LDA 0,B0T7          ;TEST CARRY PROPAGATION
01533 024072          LDA 1,KB7           ;ADDER 7 - LEFT
01534 123022          ADDZ 1,0,SZC        ;000400 + 177400
01535 101004          MOV 0,0,SZR         ;CARRY FROM ADDER 7 TO 6
01536 063077          HALT                ;IS PROBABLY BAD

01537 020160 C09:     LDA 0,B0T8          ;TEST CARRY PROPAGATION
01540 024073          LDA 1,KB8           ;ADDER 8 - LEFT
01541 123022          ADDZ 1,0,SZC        ;000200 + 177600
01542 101004          MOV 0,0,SZR         ;CHECK BCK&BCG, CARRY
01543 063077          HALT                ;OUT OF ADDER 8 INTO 7
                                          ;BOTH MUST BE HIGH

01544 020161 C10:     LDA 0,B0T9          ;TEST CARRY PROPAGATION
01545 024074          LDA 1,KB9           ;ADDER 9 - LEFT
01546 123022          ADDZ 1,0,SZC        ;000100 + 177700
01547 101004          MOV 0,0,SZR         ;CARRY FROM ADDER 9 TO 8
01550 063077          HALT                ;IS PROBABLY BAD

01551 020162 C11:     LDA 0,B0T10         ;TEST CARRY PROPAGATION
01552 024075          LDA 1,KB10          ;ADDER 10 - LEFT
01553 123022          ADDZ 1,0,SZC        ;000040 + 177740
01554 101004          MOV 0,0,SZR         ;CARRY FROM ADDER 10 TO 9
01555 063077          HALT                ;IS PROBABLY BAD

01556 020163 C12:     LDA 0,B0T11         ;TEST CARRY PROPAGATION
01557 024076          LDA 1,KB11          ;ADDER 11 - LEFT
01560 123022          ADDZ 1,0,SZC        ;000020 + 177760
01561 101004          MOV 0,0,SZR         ;CARRY FROM ADDER 11 TO 10
01562 063077          HALT                ;IS PROBABLY BAD

01563 020164 C13:     LDA 0,B0T12         ;TEST CARRY PROPAGATION
01564 024077          LDA 1,KB12          ;ADDER 12 - LEFT
01565 123022          ADDZ 1,0,SZC        ;000010 + 177770
01566 101004          MOV 0,0,SZR         ;CHECK ADDER 12 CARRY OUT
01567 063077          HALT                ;(PINS 14-15) TO ADDER
                                          ;11 CARRY IN (PINS 3-4)
                                          ;BOTH MUST BE HIGH!!

01570 020165 C14:     LDA 0,B0T13         ;TEST CARRY PROPAGATION
01571 024100          LDA 1,KB13          ;ADDER 13 - LEFT
01572 123022          ADDZ 1,0,SZC        ;000004 + 177774
01573 101004          MOV 0,0,SZR         ;CARRY FROM ADDER 13 TO 12
01574 063077          HALT                ;IS PROBABLY BAD

01575 020166 C15:     LDA 0,B0T14         ;TEST CARRY PROPAGATION
01576 024101          LDA 1,KB14          ;ADDER 14 - LEFT
01577 123022          ADDZ 1,0,SZC        ;000002 + 177776
01600 101004          MOV 0,0,SZR         ;CARRY FROM ADDER 14 TO 13
01601 063077          HALT                ;IS PROBABLY BAD
```

```
    01602 020167 C16:     LDA 0,B0T15       ;TEST CARRY PROPAGATION
    01603 024061          LDA 1,KB15        ;ADDER 15 - LEFT
    01604 123022          ADDZ 1,0,SZC      ;000001 + 177777
    01605 101004          MOV 0,0,SZR       ;CARRY FROM ADDER 15 TO 14
    01606 063077          HALT              ;IS PROBABLY BAD

    01607 020063 C17:     LDA 0,KB0         ;TEST CARRY PROPAGATION
    01610 024063          LDA 1,KB0         ;CHECK ADDER 0 CARRY OUT
    01611 107022          ADDZ 0,1,SZC      ;100000 + 100000
    01612 125004          MOV 1,1,SZR       ;CR&CG MUST BOTH =1
    01613 063077          HALT              ;INPUT TO "CRY TO AR"

    01614 020151 C18:     LDA 0,B0T1        ;TEST CARRY PROPAGATION
    01615 024064          LDA 1,KB1         ;CHECK ADDER 1
    01616 107022          ADDZ 0,1,SZC      ;040000 + 140000
    01617 125004          MOV 1,1,SZR       ;CARRY FROM ADDER 1 TO 0
    01620 063077          HALT              ;IS PROBABLY BAD

    01621 020152 C19:     LDA 0,B0T2        ;TEST CARRY PROPAGATION
    01622 024065          LDA 1,KB2         ;ADDER 2 - LEFT
    01623 107022          ADDZ 0,1,SZC      ;020000 + 160000
    01624 125004          MOV 1,1,SZR       ;CARRY FROM ADDER 2 TO 1
    01625 063077          HALT              ;IS PROBABLY BAD

    01626 020153 C20:     LDA 0,B0T3        ;TEST CARRY PROPAGATION
    01627 024066          LDA 1,KB3         ;ADDER 3 - LEFT
    01630 107022          ADDZ 0,1,SZC      ;010000 + 170000
    01631 125004          MOV 1,1,SZR       ;CARRY FROM ADDER 3 TO 2
    01632 063077          HALT              ;IS PROBABLY BAD


    01633 020154 C21:     LDA 0,B0T4        ;TEST CARRY PROPAGATION
    01634 024067          LDA 1,KB4         ;ADDER 4 - LEFT
    01635 107022          ADDZ 0,1,SZC      ;004000 + 174000
    01636 125004          MOV 1,1,SZR       ;CHECK ADDER 4 CARRY OUT
    01637 063077          HALT              ;(PINS 14-15) TO ADDER
                                            ;3 CARRY IN (PINS 3-4)
                                            ;BOTH MUST BE HIGH!!

    01640 020155 C22:     LDA 0,B0T5        ;TEST CARRY PROPAGATION
    01641 024070          LDA 1,KB5         ;ADDER 5 - LEFT
    01642 107022          ADDZ 0,1,SZC      ;002000 + 176000
    01643 125004          MOV 1,1,SZR       ;CARRY FROM ADDER 5 TO 4
    01644 063077          HALT              ;IS PROBABLY BAD

    01645 020156 C23:     LDA 0,B0T6        ;TEST CARRY PROPAGATION
    01646 024071          LDA 1,KB6         ;ADDER 6 - LEFT
    01647 107022          ADDZ 0,1,SZC      ;001000 + 177000
    01650 125004          MOV 1,1,SZR       ;CARRY FROM ADDER 6 TO 5
    01651 063077          HALT              ;IS PROBABLY BAD

    01652 020157 C24:     LDA 0,B0T7        ;TEST CARRY PROPAGATION
    01653 024072          LDA 1,KB7         ;ADDER 7 - LEFT
    01654 107022          ADDZ 0,1,SZC      ;000400 + 177400
    01655 125004          MOV 1,1,SZR       ;CARRY FROM ADDER 7 TO 6
    01656 063077          HALT              ;IS PROBABLY BAD
```

```
01657 020160 C25:    LDA 0,B0T8       ;TEST CARRY PROPAGATION
01660 024073         LDA 1,KB8        ;ADDER 8 - LEFT
01661 107022         ADDZ 0,1,SZC     ;000200 + 177600
01662 125004         MOV 1,1,SZR      ;CHECK BCK&BCG, CARRY
01663 063077         HALT             ;OUT OF ADDER 8 INTO 7
                                      ;BOTH MUST BE HIGH

01664 020161 C26:    LDA 0,B0T9       ;TEST CARRY PROPAGATION
01665 024074         LDA 1,KB9        ;ADDER 9 - LEFT
01666 107022         ADDZ 0,1,SZC     ;000100 + 177700
01667 125004         MOV 1,1,SZR      ;CARRY FROM ADDER 9 TO 8
01670 063077         HALT             ;IS PROBABLY BAD

01671 020162 C27:    LDA 0,B0T10      ;TEST CARRY PROPAGATION
01672 024075         LDA 1,KB10       ;ADDER 10 - LEFT
01673 107022         ADDZ 0,1,SZC     ;000040 + 177740
01674 125004         MOV 1,1,SZR      ;CARRY FROM ADDER 10 TO 9
01675 063077         HALT             ;IS PROBABLY BAD

01676 020163 C28:    LDA 0,B0T11      ;TEST CARRY PROPAGATION
01677 024076         LDA 1,KB11       ;ADDER 11 - LEFT
01700 107022         ADDZ 0,1,SZC     ;000020 + 177760
01701 125004         MOV 1,1,SZR      ;CARRY FROM ADDER 11 TO 10
01702 063077         HALT             ;IS PROBABLY BAD

01703 020164 C29:    LDA 0,B0T12      ;TEST CARRY PROPAGATION
01704 024077         LDA 1,KB12       ;ADDER 12 - LEFT
01705 107022         ADDZ 0,1,SZC     ;000010 + 177770
01706 125004         MOV 1,1,SZR      ;CHECK ADDER 12 CARRY OUT
01707 063077         HALT             ;(PINS 14-15) TO ADDER
                                      ;11 CARRY IN (PINS 3-4)
                                      ;BOTH MUST BE HIGH!!

01710 020165 C30:    LDA 0,B0T13      ;TEST CARRY PROPAGATION
01711 024100         LDA 1,KB13       ;ADDER 13 - LEFT
01712 107022         ADDZ 0,1,SZC     ;000004 + 177774
01713 125004         MOV 1,1,SZR      ;CARRY FROM ADDER 13 TO 12
01714 063077         HALT             ;IS PROBABLY BAD

01715 020166 C31:    LDA 0,B0T14      ;TEST CARRY PROPAGATION
01716 024101         LDA 1,KB14       ;ADDER 14 - LEFT
01717 107022         ADDZ 0,1,SZC     ;000002 + 177776
01720 125004         MOV 1,1,SZR      ;CARRY FROM ADDER 14 TO 13
01721 063077         HALT             ;IS PROBABLY BAD

01722 020167 C32:    LDA 0,B0T15      ;TEST CARRY PROPAGATION
01723 024061         LDA 1,KB15       ;ADDER 15 - LEFT
01724 107022         ADDZ 0,1,SZC     ;000001 + 177777
01725 125004         MOV 1,1,SZR      ;CARRY FROM ADDER 15 TO 14
01726 063077         HALT             ;IS PROBABLY BAD


01727 020055 C33:    LDA 0,K0         ;TEST ADDER
01730 024057         LDA 1,KONES      ;ALL 16 BITS
01731 107023         ADDZ 0,1,SNC     ; 0 + 177777
01732 124004         COM 1,1,SZR
01733 063077         HALT
```

```
01734 020055 C34:     LDA 0,K0           ;TEST ADDER
01735 024057          LDA 1,KONES        ;ALL 16 BITS
01736 107023          ADDZ 0,1,SNC       ; 177777 + 0
01737 124004          COM 1,1,SZR
01740 063077          HALT

01741 020055 C35:     LDA 0,K0           ;CHECK AND INSTRUCTION
01742 024061          LDA 1,KB15         ;CHECK AND, INST REG.
01743 123404          AND 1,0,SZR        ;DECODE (IR5,6,7)
01744 063077          HALT               ;CHECK AND INPUT TO ADDER

01745 020055 C36:     LDA 0,K0           ;CHECK AND INSTRUCTION
01746 024057          LDA 1,KONES
01747 107404          AND 0,1,SZR        ;BITS IN AC1 INDICATE
01750 063077          HALT               ;FAILING ADDERS

01751 020055 C37:     LDA 0,K0           ;CHECK AND INSTRUCTION
01752 024057          LDA 1,KONES
01753 123404          AND 1,0,SZR        ;BITS IN AC0 INDICATE
01754 063077          HALT               ;FAILING ADDERS

01755 020057 C38:     LDA 0,KONES        ;CHECK AND INSTRUCTION
01756 024057          LDA 1,KONES
01757 123423          ANDZ 1,0,SNC       ;ZERO BITS IN AC0
01760 110004          COM 0,2,SZR        ;INDICATE FAILING ADDERS
01761 063077          HALT

01762 020142 C39:     LDA 0,KOB          ;CHECK AND INSTRUCTION
01763 024141          LDA 1,KEB          ;125252&052525
01764 107404          AND 0,1,SZR        ;BITS IN AC1 INDICATE
01765 063077          HALT               ;FAILING ADDERS

01766 020142 C40:     LDA 0,KOB          ;CHECK AND INSTRUCTION
01767 024141          LDA 1,KEB          ;052525&125252
01770 123404          AND 1,0,SZR        ;BITS IN AC0 INDICATE
01771 063077          HALT               ;FAILING ADDERS

01772 020055 C41:     LDA 0,K0           ;CHECK ADC INSTRUCTION
01773 024057          LDA 1,KONES        ;IF AC0 ALL ONES CHECK
01774 122024          ADCZ 1,0,SZR       ;SX-COM.  IF AC0=1
01775 063077          HALT               ;POSSIBLE "PLUS ONE"
01776 101002          MOV 0,0,SZC        ;FAILURE.  IF CRY=1 CHECK
01777 063077          HALT               ;FOR ILLEGAL "PLUS ONE"

02000 020055 C42:     LDA 0,K0           ;CHECK ADC INSTRUCTION
02001 024055          LDA 1,K0           ; COM-ZERO + ZERO
02002 122023          ADCZ 1,0,SNC       ;SEE ABOVE
02003 110004          COM 0,2,SZR
02004 063077          HALT

02005 020142 C43:     LDA 0,KOB          ;CHECK ADC INSTRUCTION
02006 024142          LDA 1,KOB          ; COM-052525 + 052525
02007 122023          ADCZ 1,0,SNC       ;SEE ABOVE
02010 110004          COM 0,2,SZR
02011 063077          HALT

                      .EOT
```

```
 02012 020141 C44:      LDA 0,KEB          ;CHECK ADC INSTRUCTION
 02013 024141           LDA 1,KEB          ; COM-125252 + 125252
 02014 122023           ADCZ 1,0,SNC       ; SEE ABOVE
 02015 110004           COM 0,2,SZR
 02016 063077           HALT

 02017 020142 C45:      LDA 0,KOB          ;SEE ABOVE
 02020 102000           ADC 0,0
 02021 104004           COM 0,1,SZR
 02022 063077           HALT

 02023 020055 C46:      LDA 0,K0           ;CHECK INC INSTRUCTION
 02024 105400           INC 0,1            ;INC=MOV & "PLUS ONE"
 02025 131225           MOVZR 1,2,SNR      ;CHECK PLUS ONE
 02026 101003           MOV 0,0,SNC        ;AC1=RESULT OF ZERO
 02027 063077           HALT               ;INCREMENTED

 02030 020055 C47:      LDA 0,K0           ;CHECK NEG INSTRUCTION
 02031 104425           NEGZ 0,1,SNR       ;POSSIBLE FAILURES;
 02032 151003           MOV 2,2,SNC        ;SX-COM, PLUS ONE
 02033 063077           HALT

 02034 020057 C48:      LDA 0,KONES        ;CHECK NEG INSTRUCTION
 02035 104400           NEG 0,1            ;SEE ABOVE
 02036 131225           MOVZR 1,2,SNR
 02037 175003           MOV 3,3,SNC
 02040 063077           HALT

 02041 020055 C49:      LDA 0,K0           ;CHECK SUB INSTRUCTION
 02042 024055           LDA 1,K0           ; 0 - 0 = 0
 02043 122425           SUBZ 1,0,SNR       ;SUB IS A COMBINATION
 02044 151003           MOV 2,2,SNC        ;OF COM-INC-ADD
 02045 063077           HALT               ;IT SHOULD WORK!!

 02046 020055 C50:      LDA 0,K0           ;CHECK SUB INSTRUCTION
 02047 024057           LDA 1,KONES
 02050 106422           SUBZ 0,1,SZC       ;SEE ABOVE
 02051 130004           COM 1,2,SZR        ;(-1)-0=-1
 02052 063077           HALT

 02053 020055 C51:      LDA 0,K0           ;CHECK SUB INSTRUCTION
 02054 024057           LDA 1,KONES
 02055 122420           SUBZ 1,0           ;SEE ABOVE
 02056 111225           MOVZR 0,2,SNR      ; 0-(-1)=+1
 02057 175003           MOV 3,3,SNC
 02060 063077           HALT

 02061 020141 C52:      LDA 0,KEB          ;SEE ABOVE
 02062 102425           SUBZ 0,0,SNR
 02063 125003           MOV 1,1,SNC
 02064 063077           HALT
```

```
A ØØ24  .MAIN

    Ø2Ø65 126521 DØ1:     SUBZL 1,1,SKP     ;TEST RELATIVE ADDRESSING
    Ø2Ø66 ØØØØØ1          1                 ;D-M-COM PRODUCES ALL 1'S
    Ø2Ø67 Ø2Ø777          LDA Ø,.-1         ;FROM D-MULT Ø-7
    Ø2Ø7Ø 1Ø6414          SUB# Ø,1,SZR      ;D-MULT 8-15 DERIVE ALL 1'S
    Ø2Ø71 Ø63Ø77          HALT              ;FROM MB.  CHECK FOR
                                            ;CORRECT MA AT PTG3-FETCH

    Ø2Ø72 Ø2Ø4Ø2 DØ2:     LDA Ø,.+2         ;TEST RELATIVE ADDRESSING
    Ø2Ø73 126621          SUBZR 1,1,SKP     ;AS ABOVE BUT (+) DISPLACEMENT
    Ø2Ø74 1ØØØØØ          1ØØØØØ            ;D-MULTØ-7=ZEROS
    Ø2Ø75 1Ø6414          SUB# Ø,1,SZR      ;D-MULT8-15= + DISPLACEMENT
    Ø2Ø76 Ø63Ø77          HALT

    Ø2Ø77 Ø3ØØ57 DØ3:     LDA 2,KONES       ;TEST BASE REG ADDRESSING
    Ø21ØØ Ø21Ø56          LDA Ø,KØ+1,2      ;ADDRESS IS (KO+1)+(-1)=(KØ)
    Ø21Ø1 1Ø1ØØ4          MOV Ø,Ø,SZR       ;OPERAND ADDRESS IS FORMED AT
    Ø21Ø2 Ø63Ø77          HALT              ;PTG3-FETCH.  SX-MULT=ALL 1'S
                                            ;D-MULT=ADDRESS OF KØ+1, ADDER
                                            ;OUTPUT IS SENT TO MA

    Ø21Ø3 Ø3ØØ61 DØ4:     LDA 2,KB15        ;TEST BASE REG ADDRESSING
    Ø21Ø4 Ø21Ø54          LDA Ø,KØ-1,2      ;SEE ABOVE
    Ø21Ø5 1Ø1ØØ4          MOV Ø,Ø,SZR       ;(KØ-1)+(+1)=(KØ)
    Ø21Ø6 Ø63Ø77          HALT

    Ø21Ø7 Ø34147 DØ5:     LDA 3,M3          ;TEST BASE REG ADDRESSING
    Ø211Ø Ø21462          LDA Ø,KONES+3,3 ;SEE ABOVE
    Ø2111 1Ø4ØØ4          COM Ø,1,SZR       ;(KONES+3)+(-3)=(KONES)
    Ø2112 Ø63Ø77          HALT

    Ø2113 Ø341Ø3 DØ6:     LDA 3,K7          ;TEST BASE REG ADDRESSING
    Ø2114 Ø21454          LDA Ø,KBØ-7,3     ;SEE ABOVE
    Ø2115 1Ø1125          MOVZL Ø,Ø,SNR     ;(KBØ-7)+(7)=(KBØ)
    Ø2116 1Ø1ØØ3          MOV Ø,Ø,SNC
    Ø2117 Ø63Ø77          HALT

    Ø212Ø Ø2Ø142 DØ7:     LDA Ø,KOB         ;TEST STA INSTRUCTION
    Ø2121 Ø4ØØØØ          STA Ø,Ø           ;CHECK D-L-SEL IN EXE
    Ø2122 Ø24ØØØ          LDA 1,Ø           ;CYCLE.  CHECK CORRECT
    Ø2123 1Ø6414          SUB# Ø,1,SZR      ;OPERAND ADDRESS, PTG3 OF
    Ø2124 Ø63Ø77          HALT              ;FETCH. CHECK "OMIT STROBE"
                                            ;"MB LOAD"

    Ø2125 Ø24141 DØ8:     LDA 1,KEB         ;TEST STA INSTRUCTION
    Ø2126 Ø44ØØØ          STA 1,Ø           ;SEE ABOVE
    Ø2127 Ø2ØØØØ          LDA Ø,Ø
    Ø213Ø 122414          SUB# 1,Ø,SZR
    Ø2131 Ø63Ø77          HALT

    Ø2132 152121 DØ9:     ADCZL 2,2,SKP     ;TEST STA INSTRUCTION
    Ø2133 ØØØØØØ          Ø                 ;RELATIVE ADDRESSING
    Ø2134 Ø5Ø777          STA 2,.-1         ;SEE ABOVE
    Ø2135 Ø34776          LDA 3,.-2
    Ø2136 156414          SUB# 2,3,SZR
    Ø2137 Ø63Ø77          HALT
```

```
02140 034101 D10:     LDA 3,KB14         ;TEST STA INSTRUCTION
02141 054150          STA 3,PG0          ;PAGE ZERO ADDRESSING
02142 020150          LDA 0,PG0          ;SEE ABOVE
02143 116414          SUB# 0,3,SZR
02144 063077          HALT

02145 102400 D11:     SUB 0,0            ;TEST ISZ INSTRUCTION
02146 040000          STA 0,0            ;IN LOC 0, 0+1=1, NO
02147 010000          ISZ 0              ;SKIP EXPECTED.
02150 175001          MOV 3,3,SKP        ;CHECK FOR ILLEGAL INC PC
02151 063077          HALT               ;IN EXE CYCLE-PTG1
02152 024000          LDA 1,0            ;IF LOC 0 NOT=1, CHECK
02153 131225          MOVZR 1,2,SNR      ;FOR MA=0 AT PTG3-FETCH
02154 151003          MOV 2,2,SNC        ;CHECK FOR "PLUS ONE"
02155 063077          HALT               ;"MB LD EN" AND EXTEMDED PTG1
                                         ;IN EXE CYCLE

02156 102000 D12:     ADC 0,0            ;TEST ISZ INSTRUCTION
02157 040000          STA 0,0            ;LOC 0 = ALL 1'S
02160 010000          ISZ 0              ;CHECK FOR 2 "INC PC"
02161 063077          HALT               ;PTG1-PTG2 OF EXE CYCLE
02162 024000          LDA 1,0            ;IF LOC 0 NOT 0 CHECK
02163 125004          MOV 1,1,SZR        ;"PLUS ONE","MB LD EN"
02164 063077          HALT               ;EXTENDED PTG1 IN EXE CYC

02165 102400 D13:     SUB 0,0            ;TEST DSZ INSTRUCTION
02166 040000          STA 0,0            ;LOC 0=0-1=ALL 1'S, NO SKP
02167 014000          DSZ 0              ;IF SKIP, CHECK FOR ILLEGAL
02170 175001          MOV 3,3,SKP        ;INC PC AT PTG1 OF EXE CYCLE
02171 063077          HALT
02172 024000          LDA 1,0            ;IF LOC 0 NOT ALL 1'S, CHECK
02173 130004          COM 1,2,SZR        ;DURING EXE CYCLE FOR SX-COM,
02174 063077          HALT               ;ALL 1'S OUT OF SX-MULT AND
                                         ;-NO- PLUS ONE.

02175 102520 D14:     SUBZL 0,0          ;TEST DSZ INSTRUCTION
02176 040000          STA 0,0            ;LOC 0=1-1=0,  SKP
02177 014000          DSZ 0              ;CHECK FOR 2 INC PC
02200 063077          HALT               ;PTG1,PTG2 OF EXE CYCLE
02201 024000          LDA 1,0            ;IF LOC 0 NOT=0 CHECK
02202 125004          MOV 1,1,SZR        ;SX-COM,MB LD EN,SX-MULT
02203 063077          HALT               ;ALL 1'S, DURING EXE CYCLE

02204 020144 D15:     LDA 0,INDIR+1      ;TEST INDIRECT ADDRESSING
02205 040143          STA 0,INDIR        ;FETCH-DEFER-EXE CYCLES
02206 026143          LDA 1,@INDIR       ;EFA&IR5 SELECT DEFER
02207 131225          MOVZR 1,2,SNR      ;CHECK MEM OUT,MA XFR THRU
02210 175003          MOV 3,3,SNC        ;ADDER IN DEFER CYCLE
02211 063077          HALT
02212 034143          LDA 3,INDIR        ;INDIRECT ADDRESS CHANGED
02213 116414          SUB# 0,3,SZR       ;DURING DEFER CYCLE.  ILLEGAL
02214 063077          HALT               ;"AUT DEC" OR "AUT INC + DEC".
                                         ;CHECK SX-COM AND PLUS ONE
                                         ;DURING DEFER CYCLE
```

```
02215 020146 D16:     LDA 0,INDR1+1       ;TEST INDIRECT ADDRESSING
02216 040145          STA 0,INDR1         ;FETCH-DEFER-DEFER-EXF
02217 024144          LDA 1,INDIR+1       ;SAME AS PREVIOUS TEST
02220 044143          STA 1,INDIR         ;EXCEPT 2 DEFER CYCLES
02221 032145          LDA 2,@INDR1        ;SECOND DEFER RESULT
02222 155225          MOVZR 2,3,SNR       ;OF PTG1-D-MEM0
02223 175003          MOV 3,3,SNC
02224 063077          HALT
02225 030145          LDA 2,INDR1
02226 034143          LDA 3,INDIR         ;IF INDIRECT ADDRESSES ARE
02227 112415          SUB# 0,2,SNR        ;MODIFIED DURING DEFER
02230 136414          SUB# 1,3,SZR        ;PROBABLE CAUSE IS SX-COM
02231 063077          HALT                ;OR PLUS ONE

02232 102400 D17:     SUB 0,0             ;FIRST JMP INSTRUCTION
02233 126400          SUB 1,1
02234 152400          SUB 2,2
02235 176400          SUB 3,3
02236 000402          JMP .+2             ;LOAD PC FAILURE, PTG3-FETCH
02237 063077          HALT                ;EFA TO PC, CHECK JMP+JSR
02240 101005          MOV 0,0,SNR
02241 125004          MOV 1,1,SZR         ;IF AC'S ALTERED
02242 063077          HALT                ;CHECK FOR ILLEGAL
02243 151005          MOV 2,2,SNR         ;JSR OR ALC
02244 175004          MOV 3,3,SZR         ;SUBCOMMAND
02245 063077          HALT

02246 000403 D18:     JMP .+3             ;TEST JMP @
02247 002255          .+6                 ;IF JUMP FAILURE CHECK
02250 002255          .+5                 ;MEM-PC XFER THRU
02251 020776          LDA 0,.-2           ;ADDER AND LOAD PC
02252 040776          STA 0,.-2           ;AT PTG3-DEFER
02253 002775          JMP @.-3
02254 063077          HALT
02255 024772          LDA 1,.-6           ;IF INDIRECT ADDRESS CHANGED
02256 106414          SUB# 0,1,SZR        ;CHECK FOR ILLEGAL SX-COM
02257 063077          HALT                ;OR PLUS ONE IN DEFER CYC

02260 102400 D19:     SUB 0,0             ;FIRST JSR INSTRUCTION
02261 126400          SUB 1,1
02262 152400          SUB 2,2
02263 176401          SUB 3,3,SKP         ;FAILURE TO JMP ON JSR
02264 002266          .+2                 ;INSTRUCTION INDICATES
02265 004402          JSR .+2             ;"JMP+JSR" FAILED.
02266 063077          HALT                ;CHECK LOAD PC, PTG3-FETCH
02267 101005          MOV 0,0,SNR
02270 125004          MOV 1,1,SZR
02271 063077          HALT                ;IF AC0-1-2 ALTERED
02272 151004          MOV 2,2,SZR         ;CHECK AC-H-WRITE/AC-L-WRITE
02273 063077          HALT
02274 020770          LDA 0,.-10          ;IF FAILURE TO LOAD AC3
02275 116414          SUB# 0,3,SZR        ;CHECK "LOAD AR","WRITE AC"
02276 063077          HALT                ;AC-H-WRITE/AC-L-WRITE
```

```
    02277 020056 D20:     LDA 0,KONES-1    ;CHECK AUTO INDEX
    02300 040020          STA 0,20
    02301 026020          LDA 1,@20        ;AUT INC+DEC FAILED TO
    02302 106415          SUB# 0,1,SNR     ;ASSERT.  CHECK SUM 11
    02303 063077          HALT             ;ADDER=0,PTG3-FETCH

    02304 020056 D21:     LDA 0,KONES-1    ;CHECK AUTO INDEX
    02305 040021          STA 0,21
    02306 026021          LDA 1,@21        ;AUTO INDEX DIDN'T OCCUR
    02307 030021          LDA 2,21         ;PROPERLY,  CHECK FOR
    02310 112645          SUBOR 0,2,SNR    ;"PLUS ONE" AT PTG1-DEFER
    02311 151003          MOV 2,2,SNC
    02312 063077          HALT

    02313 020056 D22:     LDA 0,KONES-1    ;CHECK AUTO INDEX
    02314 040022          STA 0,22
    02315 026022          LDA 1,@22        ;OBTAINED WRONG OPERAND
    02316 130004          COM 1,2,SZR      ;CHECK OPERAND ADDRESS TO
    02317 063077          HALT             ;MA(MEM-ADDER-MEM)
                                           ;"ADDER TO MEM"M"MA LOAD"
                                           ;AT PTG3-DEFER

    02320 020060 D23:     LDA 0,KONES+1    ;CHECK AUTO DECR
    02321 040037          STA 0,37
    02322 026037          LDA 1,@37        ;AUT DEC FAILED TO SET
    02323 125225          MOVZR 1,1,SNR    ;AT PTG3-FETCH
    02324 063077          HALT

    02325 020060 D24:     LDA 0,KONES+1    ;CHECK AUT DEC
    02326 040034          STA 0,34
    02327 026034          LDA 1,@34        ;AUT DEC FAILED TO SET
    02330 106415          SUB# 0,1,SNR     ;"FORCE SX-COM"
    02331 063077          HALT

    02332 020060 D25:     LDA 0,KONES+1    ;CHECK AUT DEC
    02333 040030          STA 0,30
    02334 026030          LDA 1,@30        ;CHECK (MEM-1) TO MA
    02335 130004          COM 1,2,SZR      ;PTG3-DEFER, "MA LOAD"
    02336 063077          HALT

    02337 063500 E01:     SKPBZ 0          ;CHECK IO SKIP-IO SKIP SYNC
    02340 063077          HALT             ;GATE ON FETCH SKIP

    02341 063400 E02:     SKPBN 0          ;IO SKIP SYNC FAILURE
    02342 101001          MOV 0,0,SKP      ;CHECK IO SKIP PEND
    02343 063077          HALT

    02344 063700 E03:     SKPDZ 0          ;CHECK SELD THRU TO
    02345 063077          HALT             ;IO SKIP PEND

    02346 063600 E04:     SKPDN 0          ;IO SKIP PEND SHOULD
    02347 101001          MOV 0,0,SKP      ;GO LOW.  CHECK SELD THRU
    02350 063077          HALT             ;TO IO SKIP PEND
    02351 020003 E051:      LDA 0,3              ;IF ILLEGAL INTERRUPT HALT
    02352 040001            STA 0,1              ;AT LOC 2.
    02353 060177            NIOS 77              ;ALLOW INTERRUPTS. TEST
    02354 101000            MOV 0,0              ;FOR ION
    02355 063577            SKPBZ 77
    02356 101001            MOV 0,0,SKP          ;CHECK ION LOGIC,
    02357 063077            HALT                 ;"SELB" OC GATE, STRT
    02360 063477            SKPBN 77             ;PULSE
    02361 063077            HALT

    02362 020003 E061:      LDA 0,3              ;TEST ION RESET
    02363 040001            STA 0,1
    02364 060177            NIOS 77              ;SET ION
    02365 101000            MOV 0,0
    02366 063477            SKPBN 77             ;ION NOT ON.  SEE PREVIOUS
    02367 063077            HALT                 ;TEST
    02370 060277            NIOC 77
    02371 063577            SKPBZ 77             ;CLR DID NOT RESET
    02372 063077            HALT                 ;ION


    02373 060100 E071:      NIOS 0               ;START PULSE TO DEV 0
    02374 101000            MOV 0,0              ;SHOULD NOT SET ION
    02375 063577            SKPBZ 77
    02376 063077            HALT                 ;CPU INST ON ILLEGALLY

    02377 060077 E081:      NIO 77               ;ION SHOULD NOT SET
    02400 101000            MOV 0,0              ;WITHOUT STRT PULSE
    02401 063577            SKPBZ 77
    02402 063077            HALT                 ;CHECK ION SET LOGIC

    02403 060377 E091:      NIOP 77              ;IOPLS SHOULD HAVE
    02404 101000            MOV 0,0              ;NO AFFECT ON
    02405 063477            SKPBN 77             ;CPU STATUS
    02406 063577            SKPBZ 77
    02407 063077            HALT                 ;CHECK FOR POSSIBLE
    02410 063677            SKPDN 77             ;ILLEGAL
    02411 063777            SKPDZ 77             ;START PULSE
    02412 063077            HALT

    02413 060477 E101:      READS 0              ;INSURE THAT DIA INSTRUCTION
    02414 111000            MOV 0,2              ;DOES NOT HANG UP THE CPU
    02415 112414            SUB# 0,2,SZR
    02416 063077            HALT

    02417 062677 E111:      IORST                ;CHECK CLEAR IO
    02420 126001            ADC 1,1,SKP
    02421 063077            HALT                 ;IORST CAUSED SKIP!!!
    02422 065477            INTA 1               ;INTA SHOULD READ
    02423 125004            MOV 1,1,SZR          ;ZEROS FROM IO DATA
    02424 063077            HALT                 ;BUS.  CHECK "READ IO",
                                                 ;"WRITE AC"
    02425 102620             SUBZR 0,0
    02426 152440             SUBO 2,2
    02427 073101             DIV
    02430 101002             MOV 0,0,SZC        ;IS MUL/DIV PRESENT ?
    02431 000406             JMP CKMD           ;YES, GO CHECK IT.



    02432 034045             LDA 3,45           ;MOTHER HEN STUFF
    02433 011403             ISZ 3,3
    02434 000401             JMP .+1
    02435 000376             JMP LOOP

    02436 000376             LOOP
    02437 020777 CKMD:       LDA 0,.-1
    02440 000403             JMP MULDIV+1

    02441 002442             .+1
    02442 020777 MULDIV:     LDA 0,.-1
    02443 040170             STA 0,LADDR        ;LOOP ADDRESS

                             .EOT
```

```
                ;       TEST FOR THE PRESENCE OF MUL/DIV LOGIC
                ;       AND VERIFY THE OPERATION OF THE DIVIDE REJECT.
                ;       DEFINATION OF THE DIVIDE IS AC0-1/AC2.  IF
                ;       AC0 > OR = AC2 THE DIVIDE IS NOT PERFORMED AND
                ;       THE OPERATION TERMINATES AFTER ONE CYCLE (THE DIV
                ;       FIRST CYCLE).  CYCLE SEQUENCE BEGINNING WITH THE
                ;       DIV INSTRUCTION IS; FETCH-EXECUTE-DIV FIRST(STATE
                ;       SUPPRESS ON)-NEXT FETCH CYCLE.

                ;       IN THE FOLLOWING TESTS AC0 IS = OR > AC2.
                ;       "OVFLO TO DIV" SHOULD OCCUR DURING "DIV FIRST"
                ;       RESETTING MUL+DIV AND TERMINATING THE DIVIDE
                ;       INSTRUCTION WITH CRY=1.


 02444 102400 F01:      SUB 0,0             ;SEE ABOVE DISCUSSION
 02445 126400           SUB 1,1             ;THIS IS THE FIRST TEST
 02446 152400           SUB 2,2             ;OF MUL/DIV LOGIC.  CHECK
 02447 176440           SUBO 3,3            ;"MUL+DIV DECODE", "MUL+DIV"
 02450 073101           DIV                 ;"DIV FIRST", "OVFLO TO DIV"
 02451 101003           MOV 0,0,SNC         ;DURING "DIV FIRST" CYCLE
 02452 063077           HALT

 02453 102620 F02:      SUBZR 0,0           ;SEE ABOVE
 02454 126620           SUBZR 1,1
 02455 152620           SUBZR 2,2
 02456 176620           SUBZR 3,3
 02457 073101           DIV
 02460 101003           MOV 0,0,SNC
 02461 063077           HALT

 02462 102620 F03:      SUBZR 0,0           ;CHECK FOR POSSIBLE
 02463 152620           SUBZR 2,2           ;AC SELECTION ERROR.
 02464 176400           SUB 3,3
 02465 126440           SUBO 1,1            ;SOURCE AC IS 2
 02466 073101           DIV                 ;DESTINATION AC IS 0
 02467 125003           MOV 1,1,SNC
 02470 063077           HALT

 02471 102000 F04:      ADC 0,0             ;SEE PREVIOUS TEST
 02472 152000           ADC 2,2
 02473 176620           SUBZR 3,3
 02474 126620           SUBZR 1,1
 02475 073101           DIV
 02476 125003           MOV 1,1,SNC
 02477 063077           HALT

 02500 102400 F05:      SUB 0,0             ;MUL/DIV LOGIC SELECTED
 02501 152440           SUBO 2,2            ;WITH DICC, INSTEAD
 02502 073201           DOCC 2,1            ;OF DOCS.  CHECK MB8-9
 02503 101002           MOV 0,0,SZC         ;GATE IF "MUL+DIV DECODE"
 02504 063077           HALT                ;LOGIC

 02505 102400 F06:      SUB 0,0             ;MUL/DIV LOGIC SELECTED
 02506 152440           SUBO 2,2            ;WITH DOC INSTEAD
 02507 073001           DOC 2,1             ;OF DOCS.  CHECK MB9
 02510 101002           MOV 0,0,SZC         ;GATE OF "MUL+DIV DECODE"
 02511 063077           HALT                ;LOGIC
```

```
A 0031  .MAIN
   02512 102400 F07:     SUB 0,0
   02513 152440          SUBO 2,2
   02514 073501          073501            ;SKPBZ 01
   02515 063077          HALT              ;SKIP FAILURE (PROCESSOR)
   02516 101002          MOV 0,0,SZC       ;"MUL+DIV DECODE" ON ILLEGALLY
   02517 063077          HALT              ;CHECK MB7 GATE

   02520 102400 F08:     SUB 0,0           ;DOAS SELECTED MUL/DIV
   02521 152440          SUBO 2,2          ;INSTEAD OF DOCS.  CHECK
   02522 071101          DOAS 2,1          ;MB5 GATE OF "MUL+DIV DECODE"
   02523 101002          MOV 0,0,SZC       ;LOGIC
   02524 063077          HALT

   02525 102400 F09:     SUB 0,0           ;DOBS SELECTED MUL/DIV
   02526 152440          SUBO 2,2          ;INSTEAD OF DOCS.  CHECK
   02527 072101          DOBS 2,1          ;MB6 GATE OF "MUL+DIV
   02530 101002          MOV 0,0,SZC       ;DECODE" LOGIC

   02531 102400 F10:     SUB 0,0           ;CHECK FOR ILLEGAL MUL/DIV
   02532 152440          SUBO 2,2          ;SELECT DUE TO DEVICE CODE
   02533 073100          DOCS 2,0          ;RECOGNITION FAILURE
   02534 062677          IORST             ;(MB10-15)
   02535 101002          MOV 0,0,SZC       ;TEST MB15
   02536 063077          HALT

   02537 102400 F11:     SUB 0,0           ;SEE ABOVE
   02540 152440          SUBO 2,2
   02541 073103          DOCS 2,3          ;TEST MB14
   02542 062677          IORST
   02543 101002          MOV 0,0,SZC
   02544 063077          HALT

   02545 102400 F12:     SUB 0,0           ;SEE ABOVE
   02546 152440          SUBO 2,2
   02547 073105          DOCS 2,5          ;TEST MB13
   02550 062677          IORST
   02551 101002          MOV 0,0,SZC
   02552 063077          HALT

   02553 102400 F13:     SUB 0,0           ;SEE ABOVE
   02554 152440          SUBO 2,2
   02555 073111          DOCS 2,11         ;TEST MB12
   02556 062677          IORST
   02557 101002          MOV 0,0,SZC
   02560 063077          HALT

   02561 102400 F14:     SUB 0,0           ;SEE ABOVE
   02562 152440          SUBO 2,2
   02563 073121          DOCS 2,21         ;TEST MB11
   02564 062677          IORST
   02565 101002          MOV 0,0,SZC
   02566 063077          HALT
```

```
02567 102400 F15:     SUB 0,0            ;SEE ABOVE
02570 152440          SUBO 2,2
02571 072541          DICS 2,41          ;TEST MB10
02572 062677          IORST
02573 101002          MOV 0,0,SZC
02574 063077          HALT

02575 102400 F16:     SUB 0,0            ;TEST "MUL+DIV DECODE"
02576 152440          SUBO 2,2           ;AC DECODE, MB3-4
02577 063101          DOCS 0,1
02600 101002          MOV 0,0,SZC        ;DIVIDE = DOCS 2,1 ONLY!!
02601 063077          HALT

02602 102400 F17:     SUB 0,0            ;SEE ABOVE
02603 152440          SUBO 2,2
02604 077101          DOCS 3,1
02605 101002          MOV 0,0,SZC
02606 063077          HALT

02607 102400 F18:     SUB 0,0,           ;SEE ABOVE
02610 152440          SUBO 2,2
02611 067101          DOCS 1,1
02612 101002          MOV 0,0,SZC
02613 063077          HALT

02614 102000 F18A:    ADC 0,0            ;INSURE THAT ONLY MUL/DIV
02615 060401          DIA 0,1            ;OPERATIONS ARE ALLOWED
02616 100014          COM# 0,0,SZR       ;FOR DEVICE CODE "01"
02617 063077          HALT               ;DIA 0,1 = NO OPERATION

02620 020106 F18B:    LDA 0,K77          ;SAME AS ABOVE
02621 105000          MOV 0,1            ;"CODE 01" & "PTG3"
02622 065401          DIB 1,1            ;BLOCK "AC WRITE"
02623 106414          SUB# 0,1,SZR
02624 063077          HALT
```

```
                          ;          THE FOLLOWING ARE DIVIDE TESTS CHOSEN TO
                          ;          PRODUCE QUOTIENTS OF ZERO AND VARIOUS
                          ;          REMAINDERS.  MQ FAILURES SHOULD BE SUSPECTED.

                          ;          THIS IS THE FIRST COMPLETE DIVIDE.  CHECK THE
                          ;          OVERALL CYCLING. ---

                          ;          1 - DIV FIRST
                          ;          8 - CYCLE
                          ;          1 - LAST



02625 102440 F19:         SUBO 0,0               ;(0-1)/2 = 00, R1
02626 024061              LDA 1,K1               ;IF HANGUP, "CYCLE" MAY NOT HAVE
02627 131400              INC 1,2                ;SET OR "MUL DIV TC" MAY HAVE
02630 073101              DIV                    ;FAILED TO ASSERT.
02631 101002              MOV 0,0,SZC            ;DIVIDE ABORT.  ILLEGAL
02632 063077              HALT                   ;  "OVFLO TO DIV"
02633 125004              MOV 1,1,SZR            ;WRONG QUOTIENT (AC1)
02634 063077              HALT                   ;SHOULD BE ZERO
02635 101225              MOVZR 0,0,SNR
02636 101003              MOV 0,0,SNC            ;WRONG REMAINDER
02637 063077              HALT                   ;SHOULD BE 1

02640 102440 F20:         SUBO 0,0               ;(0-17)/20 = 00, R17
02641 024104              LDA 1,K17              ;SEE ABOVE
02642 131400              INC 1,2                ;SUSPECT MQ12-15
02643 073101              DIV
02644 101002              MOV 0,0,SZC            ;DIVIDE ABORT.
02645 063077              HALT                   ;ILLEGAL "OVFLO TO DIV"
02646 125004              MOV 1,1,SZR            ;WRONG QUOTIENT (AC1)
02647 063077              HALT                   ;SHOULD BE ZERO
02650 034104              LDA 3,K17
02651 162414              SUB# 3,0,SZR           ;REMAONDER WRONG (AC0)
02652 063077              HALT                   ;SHOULD BE 17
```

```
    02653 102440 F21:      SUBO 0,0            ;(0-377)/400 = Q0. R377
    02654 024110           LDA 1,K377          ;SEE ABOVE
    02655 131400           INC 1,2             ;SUSPECT MQ8-11
    02656 073101           DIV
    02657 101002           MOV 0,0,SZC         ;DIVIDE ABORT
    02660 063077           HALT                ;ILLEGAL "OVFLO TO DIV"
    02661 125004           MOV 1,1,SZR         ;WRONG QUOTIENT (AC1)
    02662 063077           HALT                ;SHOULD BE ZERO
    02663 034110           LDA 3,K377
    02664 162414           SUB# 3,0,SZR        ;REMAINDER WRONG (AC0)
    02665 063077           HALT                ;SHOULD BE 377

    02666 102440 F22:      SUBO 0,0            ;(0-7777)/10000 = Q0, R7777
    02667 024114           LDA 1,K7777         ;SEE ABOVE
    02670 131400           INC 1,2             ;SUSPECT MQ4-7
    02671 073101           DIV
    02672 101002           MOV 0,0,SZC         ;DIVIDE ABORT
    02673 063077           HALT                ;ILLEGAL "OVFLO TO DIV"
    02674 125004           MOV 1,1,SZR         ;WRONG QUOTIENT (AC1)
    02675 063077           HALT                ;SHOULD BE ZERO
    02676 034114           LDA 3,K7777
    02677 162414           SUB# 3,0,SZR        ;REMAINDER WRONG (AC0)
    02700 063077           HALT                ;SHOULD BE 7777

    02701 102440 F23:      SUBO 0,0            ;(0-177776)/177777 = Q0, R177776
    02702 126120           ADCZL 1,1           ;SEE ABOVE
    02703 152000           ADC 2,2             ;SUSPECT MQ 0-3
    02704 073101           DIV
    02705 101002           MOV 0,0,SZC         ;DIVIDE ABORT
    02706 063077           HALT                ;ILLEGAL "OVFLO TO DIV"
    02707 125004           MOV 1,1,SZR         ;WRONG QUOTIENT (AC1)
    02710 063077           HALT                ;SHOULD BE ZERO
    02711 114225           COMZR 0,3,SNR
    02712 175003           MOV 3,3,SNC         ;WRONG REMAINDER (AC0)
    02713 063077           HALT                ;SHOULD BE 177776

                           ;THE FOLLOWING 16 DIVIDE TESTS PRODUCE SUCCESSIVE
                           ;QUOTIENTS OF 1,3,7,17, ETC., TO 177777.  ALL
                           ;REMAINDERS ARE ZERO.  THIS IS THE FIRST ATTEMPT
                           ;TO PRODUCE NON-ZERO QUOTIENTS.  CHECK "CRY TO
                           ;AR", "SHIFT AR LEFT".  QUOTIENT IS GENERATED
                           ;IN THE AR BY SUCCESSIVE LEFT SHIFTS.


    02714 102440 F24:      SUBO 0,0            ;SEE ABOVE
    02715 126520           SUBZL 1,1           ; (0-1)/1 = Q=1, R=0
    02716 152520           SUBZL 2,2
    02717 073101           DIV
    02720 101002           MOV 0,0,SZC         ;DIVIDE ABORT ILLEGAL
    02721 063077           HALT                ;CHECK "OVFLO TO DIV"
    02722 101004           MOV 0,0,SZR         ;REMAINDER WRONG (AC0)
    02723 063077           HALT                ;SHOULD BE ZERO
    02724 176520           SUBZL 3,3
    02725 136414           SUB# 1,3,SZR        ;QUOTIENT WRONG (AC1)
    02726 063077           HALT                ;SHOULD BE 1
```

```
 02727 102440 F25:     SUBO 0,0           ;SEE ABOVE
 02730 024102          LDA 1,K3           ; (0-3)/1 = Q=3, R=0
 02731 152520          SUBZL 2,2
 02732 073101          DIV
 02733 101002          MOV 0,0,SZC        ;DIVIDE ABORT ILLEGAL
 02734 063077          HALT               ;CHECK "OVFLO TO DIV"
 02735 101004          MOV 0,0,SZR        ;REMAINDER WRONG (AC0)
 02736 063077          HALT               ;SHOULD BE ZERO
 02737 034102          LDA 3,K3
 02740 136414          SUB# 1,3,SZR       ;QUOTIENT WRONG (AC1)
 02741 063077          HALT               ;SHOULD BE 3

 02742 102440 F26:     SUBO 0,0           ;SEE ABOVE
 02743 024103          LDA 1,K7           ; (0-7)/1 = Q=7, R=0
 02744 152520          SUBZL 2,2
 02745 073101          DIV
 02746 101002          MOV 0,0,SZC        ;DIVIDE ABORT ILLEGAL
 02747 063077          HALT               ;CHECK "OVFLO TO DIV"
 02750 101004          MOV 0,0,SZR        ;REMAINDER WRONG (AC0)
 02751 063077          HALT               ;SHOULD BE ZERO
 02752 034103          LDA 3,K7
 02753 136414          SUB# 1,3,SZR       ;QUOTIENT WRONG (AC1)
 02754 063077          HALT               ;SHOULD BE 7

 02755 102440 F27:     SUBO 0,0           ;SEE ABOVE
 02756 024104          LDA 1,K17          ; (0-17)/1 = Q=17, R=0
 02757 152520          SUBZL 2,2
 02760 073101          DIV
 02761 101002          MOV 0,0,SZC        ;DIVIDE ABORT ILLEGAL
 02762 063077          HALT               ;CHECK "OVFLO TO DIV"
 02763 101004          MOV 0,0,SZR        ;REMAINDER WRONG (AC0)
 02764 063077          HALT               ;SHOULD BE ZERO
 02765 034104          LDA 3,K17
 02766 136414          SUB# 1,3,SZR       ;QUOTIENT WRONG (AC1)
 02767 063077          HALT               ;SHOULD BE 17

 02770 102440 F28:     SUBO 0,0           ;SEE ABOVE
 02771 024105          LDA 1,K37          ; (0-37)/1 = Q=37, R=0
 02772 152520          SUBZL 2,2
 02773 073101          DIV
 02774 101002          MOV 0,0,SZC        ;DIVIDE ABORT ILLEGAL
 02775 063077          HALT               ;CHECK "OVFLO TO DIV"
 02776 101004          MOV 0,0,SZR        ;REMAINDER WRONG (AC0)
 02777 063077          HALT               ;SHOULD BE ZERO
 03000 034105          LDA 3,K37
 03001 136414          SUB# 1,3,SZR       ;QUOTIENT WRONG (AC1)
 03002 063077          HALT               ;SHOULD BE 37
```

```
03003 102440 F29:     SUBO 0,0           ;SEE ABOVE
03004 024106          LDA 1,K77          ; (0-77)/1 = Q=77, R=0
03005 152520          SUBZL 2,2
03006 073101          DIV
03007 101002          MOV 0,0,SZC        ;DIVIDE ABORT ILLEGAL
03010 063077          HALT               ;CHECK "OVFLO TO DIV"
03011 101004          MOV 0,0,SZR        ;REMAINDER WRONG (AC0)
03012 063077          HALT               ;SHOULD BE ZERO
03013 034106          LDA 3,K77
03014 136414          SUB# 1,3,SZR       ;QUOTIENT WRONG (AC1)
03015 063077          HALT               ;SHOULD BE 77


03016 102440 F30:     SUBO 0,0           ;SEE ABOVE
03017 024107          LDA 1,K177         ; (0-177)/1 = Q=177, R=0
03020 152520          SUBZL 2,2
03021 073101          DIV
03022 101002          MOV 0,0,SZC        ;DIVIDE ABORT ILLEGAL
03023 063077          HALT               ;CHECK "OVFLO TO DIV"
03024 101004          MOV 0,0,SZR        ;REMAINDER WRONG (AC0)
03025 063077          HALT               ;SHOULD BE ZERO
03026 034107          LDA 3,K177
03027 136414          SUB# 1,3,SZR       ;QUOTIENT WRONG (AC1)
03030 063077          HALT               ;SHOULD BE 177


03031 102440 F31:     SUBO 0,0           ;SEE ABOVE
03032 024110          LDA 1,K377         ; (0-377)/1 = Q=377, R=0
03033 152520          SUBZL 2,2
03034 073101          DIV
03035 101002          MOV 0,0,SZC        ;DIVIDE ABORT ILLEGAL
03036 063077          HALT               ;CHECK "OVFLO TO DIV"
03037 101004          MOV 0,0,SZR        ;REMAINDER WRONG (AC0)
03040 063077          HALT               ;SHOULD BE ZERO
03041 034110          LDA 3,K377
03042 136414          SUB# 1,3,SZR       ;QUOTIENT WRONG (AC1)
03043 063077          HALT               ;SHOULD BE 377

03044 102440 F32:     SUBO 0,0           ;SEE ABOVE
03045 024111          LDA 1,K777         ; (0-777)/1 = Q=777, R=0
03046 152520          SUBZL 2,2
03047 073101          DIV
03050 101002          MOV 0,0,SZC        ;DIVIDE ABORT ILLEGAL
03051 063077          HALT               ;CHECK "OVFLO TO DIV"
03052 101004          MOV 0,0,SZR        ;REMAINDER WRONG (AC0)
03053 063077          HALT               ;SHOULD BE ZERO
03054 034111          LDA 3,K777
03055 136414          SUB# 1,3,SZR       ;QUOTIENT WRONG (AC1)
03056 063077          HALT               ;SHOULD BE 777
```

```
03057 102440 F33:     SUBO 0,0          ;SEE ABOVE
03060 024112          LDA 1,K1777       ; (0-1777)/1 = Q=1777, R=0
03061 152520          SUBZL 2,2
03062 073101          DIV
03063 101002          MOV 0,0,SZC       ;DIVIDE ABORT ILLEGAL
03064 063077          HALT              ;CHECK "OVFLO TO DIV"
03065 101004          MOV 0,0,SZR       ;REMAINDER WRONG (AC0)
03066 063077          HALT              ;SHOULD BE ZERO
03067 034112          LDA 3,K1777
03070 136414          SUB# 1,3,SZR      ;QUOTIENT WRONG (AC1)
03071 063077          HALT              ;SHOULD BE 1777

03072 102440 F34:     SUBO 0,0          ;SEE ABOVE
03073 024113          LDA 1,K3777       ; (0-3777)/1 = Q=3777, R=0
03074 152520          SUBZL 2,2
03075 073101          DIV
03076 101002          MOV 0,0,SZC       ;DIVIDE ABORT ILLEGAL
03077 063077          HALT              ;CHECK "OVFLO TO DIV"
03100 101004          MOV 0,0,SZR       ;REMAINDER WRONG (AC0)
03101 063077          HALT              ;SHOULD BE ZERO
03102 034113          LDA 3,K3777
03103 136414          SUB# 1,3,SZR      ;QUOTIENT WRONG (AC1)
03104 063077          HALT              ;SHOULD BE 3777

03105 102440 F35:     SUBO 0,0          ;SEE ABOVE
03106 024114          LDA 1,K7777       ; (0-7777)/1 = Q=7777, R=0
03107 152520          SUBZL 2,2
03110 073101          DIV
03111 101002          MOV 0,0,SZC       ;DIVIDE ABORT ILLEGAL
03112 063077          HALT              ;CHECK "OVFLO TO DIV"
03113 101004          MOV 0,0,SZR       ;REMAINDER WRONG (AC0)
03114 063077          HALT              ;SHOULD BE ZERO
03115 034114          LDA 3,K7777
03116 136414          SUB# 1,3,SZR      ;QUOTIENT WRONG (AC1)
03117 063077          HALT              ;SHOULD BE 7777

03120 102440 F36:     SUBO 0,0          ;SEE ABOVE
03121 024115          LDA 1,K1.4K       ; (0-17777)/1 = Q=17777, R=0
03122 152520          SUBZL 2,2
03123 073101          DIV
03124 101002          MOV 0,0,SZC       ;DIVIDE ABORT ILLEGAL
03125 063077          HALT              ;CHECK "OVFLO TO DIV"
03126 101004          MOV 0,0,SZR       ;REMAINDER WRONG (AC0)
03127 063077          HALT              ;SHOULD BE ZERO
03130 034115          LDA 3,K1.4K
03131 136414          SUB# 1,3,SZR      ;QUOTIENT WRONG (AC1)
03132 063077          HALT              ;SHOULD BE 17777
```

```
03133 102440 F37:    SUBO 0,0          ;SEE ABOVE
03134 024116         LDA 1,K3.4K       ; (0-37777)/1 = Q=37777, R=0
03135 152520         SUBZL 2,2
03136 073101         DIV
03137 101002         MOV 0,0,SZC       ;DIVIDE ABORT ILLEGAL
03140 063077         HALT              ;CHECK "OVFLO TO DIV"
03141 101004         MOV 0,0,SZR       ;REMAINDER WRONG (AC0)
03142 063077         HALT              ;SHOULD BE ZERO
03143 034116         LDA 3,K3.4K
03144 136414         SUB# 1,3,SZR      ;QUOTIENT WRONG (AC1)
03145 063077         HALT              ;SHOULD BE 37777

03146 102440 F38:    SUBO 0,0          ;SEE ABOVE
03147 024117         LDA 1,K7.4K       ; (0-77777)/1 = Q=77777, R=0
03150 152520         SUBZL 2,2
03151 073101         DIV
03152 101002         MOV 0,0,SZC       ;DIVIDE ABORT ILLEGAL
03153 063077         HALT              ;CHECK "OVFLO TO DIV"
03154 101004         MOV 0,0,SZR       ;REMAINDER WRONG (AC0)
03155 063077         HALT              ;SHOULD BE ZERO
03156 034117         LDA 3,K7.4K
03157 136414         SUB# 1,3,SZR      ;QUOTIENT WRONG (AC1)
03160 063077         HALT              ;SHOULD BE 77777

03161 102440 F39:    SUBO 0,0          ;SEE ABOVE
03162 024120         LDA 1,K17.4K      ; (0-177777)/1 = Q=177777, R=0
03163 152520         SUBZL 2,2
03164 073101         DIV
03165 101002         MOV 0,0,SZC       ;DIVIDE ABORT ILLEGAL
03166 063077         HALT              ;CHECK "OVFLO TO DIV"
03167 101004         MOV 0,0,SZR       ;REMAINDER WRONG (AC0)
03170 063077         HALT              ;SHOULD BE ZERO
03171 034120         LDA 3,K17.4K
03172 136414         SUB# 1,3,SZR      ;QUOTIENT WRONG (AC1)
03173 063077         HALT              ;SHOULD BE 177777


03174 102400 G01:    SUB 0,0           ;(0*1) + 0 = 0
03175 152400         SUB 2,2           ;THIS IS THE FIRST MULTIPLY
03176 126520         SUBZL 1,1         ;IF HANGUP CHECK "MUL+DIV"
03177 073301         MUL               ;AND CORRECT CYCLING
                                       ;        1 "MUL FIRST"
                                       ;        8 "MUL CYCLE"
                                       ;        1 "MUL+DIV LAST"

03200 125004         MOV 1,1,SZR
03201 063077         HALT              ;AC1 NOT 0.  IF AC1=1
                                       ;MULTIPLY APPARENTLY DID
                                       ;NOT TAKE PLACE.  CHECK
                                       ;"MUL+DIV DECODE"

03202 102400 G02:    SUB 0,0           ;SEE ABOVE
03203 152400         SUB 2,2           ;(0*1)+0=0
03204 126520         SUBZL 1,1
03205 073301         MUL               ;AC2 SHOULD NEVER BE
03206 101005         MOV 0,0,SNR       ;CHANGED DURING A MUL
03207 151004         MOV 2,2,SZR       ;AC0 SHOULD BE ZERO
03210 063077         HALT              ;(HIGH ORDER PRODUCT)
```

```
                          ;THE FOLLOWING 16 TESTS CHECK FOR CORRECT
                          ;LOADING OF THE MQ REGISTER DURING A
                          ;MULTIPLY. (AC1*AC2)+AC0 = AC0-1

03211 102400 G03:         SUB 0,0            ;SEE ABOVE
03212 152520              SUBZL 2,2          ;(100000*1)+0 = 100000
03213 024063              LDA 1,KB0          ;CHECK MQ SHIFT PULSE
03214 073301              MUL                ;AND SUM 15 IN LAST
03215 034063              LDA 3,KB0          ;MUL CYCLE
03216 101005              MOV 0,0,SNR
03217 166414              SUB# 3,1,SZR       ;SUSPCET MQ0
03220 063077              HALT

03221 102400 G04:         SUB 0,0            ;SEE ABOVE
03222 152520              SUBZL 2,2          ;(140000*1)+0 = 140000
03223 024151              LDA 1,B0T1
03224 073301              MUL
03225 034151              LDA 3,B0T1         ;SUSPECT MQ1
03226 101005              MOV 0,0,SNR
03227 166414              SUB# 3,1,SZR
03230 063077              HALT

03231 102400 G05:         SUB 0,0            ;SEE ABOVE
03232 152520              SUBZL 2,2          ;(160000*1)+0 = 160000
03233 024152              LDA 1,B0T2
03234 073301              MUL
03235 034152              LDA 3,B0T2         ;SUSPECT MQ2
03236 101005              MOV 0,0,SNR
03237 166414              SUB# 3,1,SZR
03240 063077              HALT

03241 102400 G06:         SUB 0,0            ;SEE ABOVE
03242 152520              SUBZL 2,2          ;(170000*1)+0 = 170000
03243 024153              LDA 1,B0T3
03244 073301              MUL
03245 034153              LDA 3,B0T3         ;SUSPECT MQ3
03246 101005              MOV 0,0,SNR
03247 166414              SUB# 3,1,SZR
03250 063077              HALT

03251 102400 G07:         SUB 0,0            ;SEE ABOVE
03252 152520              SUBZL 2,2          ;(174000*1)+0 = 174000
03253 024154              LDA 1,B0T4
03254 073301              MUL
03255 034154              LDA 3,B0T4         ;SUSPECT MQ4
03256 101005              MOV 0,0,SNR
03257 166414              SUB# 3,1,SZR
03260 063077              HALT

03261 102400 G08:         SUB 0,0            ;SEE ABOVE
03262 152520              SUBZL 2,2          ;(176000*1)+0 = 176000
03263 024155              LDA 1,B0T5
03264 073301              MUL
03265 034155              LDA 3,B0T5         ;SUSPECT MQ5
03266 101005              MOV 0,0,SNR
03267 166414              SUB# 3,1,SZR
03270 063077              HALT
```

```
03271 102400 G09:      SUB 0,0             ;SEE ABOVE
03272 152520           SUBZL 2,2           ;(177000*1)+0 = 177000
03273 024156           LDA 1,B0T6
03274 073301           MUL
03275 034156           LDA 3,B0T6          ;SUSPECT MQ6
03276 101005           MOV 0,0,SNR
03277 166414           SUB# 3,1,SZR
03300 063077           HALT

03301 102400 G10:      SUB 0,0             ;SEE ABOVE
03302 152520           SUBZL 2,2           ;(177400*1)+0 = 177400
03303 024157           LDA 1,B0T7
03304 073301           MUL
03305 034157           LDA 3,B0T7          ;SUSPECT MQ7
03306 101005           MOV 0,0,SNR
03307 166414           SUB# 3,1,SZR
03310 063077           HALT

03311 102400 G11:      SUB 0,0             ;SEE ABOVE
03312 152520           SUBZL 2,2           ;(177600*1)+0 = 177600
03313 024160           LDA 1,B0T8
03314 073301           MUL
03315 034160           LDA 3,B0T8          ;SUSPECT MQ8
03316 101005           MOV 0,0,SNR
03317 166414           SUB# 3,1,SZR
03320 063077           HALT

03321 102400 G12:      SUB 0,0             ;SEE ABOVE
03322 152520           SUBZL 2,2           ;(177700*1)+0 = 177700
03323 024161           LDA 1,B0T9
03324 073301           MUL
03325 034161           LDA 3,B0T9          ;SUSPECT MQ9
03326 101005           MOV 0,0,SNR
03327 166414           SUB# 3,1,SZR
03330 063077           HALT

03331 102400 G13:      SUB 0,0             ;SEE ABOVE
03332 152520           SUBZL 2,2           ;(177740*1)+0 = 177740
03333 024162           LDA 1,B0T10
03334 073301           MUL
03335 034162           LDA 3,B0T10         ;SUSPECT MQ10
03336 101005           MOV 0,0,SNR
03337 166414           SUB# 3,1,SZR
03340 063077           HALT

03341 102400 G14:      SUB 0,0             ;SEE ABOVE
03342 152520           SUBZL 2,2           ;(177760*1)+0 = 177760
03343 024163           LDA 1,B0T11
03344 073301           MUL
03345 034163           LDA 3,B0T11         ;SUSPECT MQ11
03346 101005           MOV 0,0,SNR
03347 166414           SUB# 3,1,SZR
03350 063077           HALT
```

```
03351 102400 G15:      SUB 0,0              ;SEE ABOVE
03352 152520           SUBZL 2,2            ;(177770*1)+0 = 177770
03353 024164           LDA 1,B0T12
03354 073301           MUL
03355 034164           LDA 3,B0T12          ;SUSPECT MQ12
03356 101005           MOV 0,0,SNR
03357 166414           SUB# 3,1,SZR
03360 063077           HALT

03361 102400 G16:      SUB 0,0              ;SEE ABOVE
03362 152520           SUBZL 2,2            ;(177774*1)+0 = 177774
03363 024165           LDA 1,B0T13
03364 073301           MUL
03365 034165           LDA 3,B0T13          ;SUSPECT MQ13
03366 101005           MOV 0,0,SNR
03367 166414           SUB# 3,1,SZR
03370 063077           HALT

03371 102400 G17:      SUB 0,0              ;SEE ABOVE
03372 152520           SUBZL 2,2            ;(177776*1)+0 = 177776
03373 024166           LDA 1,B0T14
03374 073301           MUL
03375 034166           LDA 3,B0T14          ;SUSPECT MQ14
03376 101005           MOV 0,0,SNR
03377 166414           SUB# 3,1,SZR
03400 063077           HALT

03401 102400 G18:      SUB 0,0              ;SEE ABOVE
03402 152520           SUBZL 2,2            ;(177777*1)+0 = 177777
03403 024167           LDA 1,B0T15
03404 073301           MUL
03405 034167           LDA 3,B0T15          ;SUSPECT MQ15
03406 101005           MOV 0,0,SNR
03407 166414           SUB# 3,1,SZR
03410 063077           HALT

03411 102400 G19:      SUB 0,0              ;CHECK TO SEE THAT
03412 152520           SUBZL 2,2            ;AC2 IS NOT
03413 024101           LDA 1,KB14           ;ALTERED DURING
03414 073301           MUL                  ;A MUL INSTRUCTION
03415 151225           MOVZR 2,2,SNR
03416 151003           MOV 2,2,SNC
03417 063077           HALT

03420 102520 G20:      SUBZL 0,0            ;CHECK TO SEE THAT
03421 152520           SUBZL 2,2            ;(AC0) IS ADDED TO PRODUCT
03422 024061           LDA 1,KB15           ;AT THE END OF A MUL
03423 073301           MUL
03424 034101           LDA 3,KB14           ;(1*1)+1=2
03425 101005           MOV 0,0,SNR
03426 136414           SUB# 1,3,SZR
03427 063077           HALT
```

```
    03430 102000 G211      ADC 0,0           ;SAME AS ABOVE
    03431 152520           SUBZL 2,2
    03432 126520           SUBZL 1,1
    03433 073301           MUL               ;(1*1)+177777=000001-000000
    03434 034061           LDA 3,KB15
    03435 125005           MOV 1,1,SNR
    03436 116414           SUB# 0,3,SZR
    03437 063077           HALT

    03440 034045           LDA 3,45          ;MOTHER HEN STUFF
    03441 011403           ISZ 3,3
    03442 000401           JMP .+1
    03443 002170           JMP @LADDR

                           .END
```