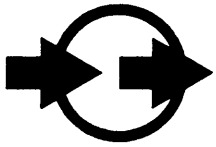
DC-220
DIAGNOSTIC AND RELIABILITY
SOFTWARE MANUAL
PN 220005



western peripherals

2893 EAST LA PALMA AVENUE
ANAHEIM, CALIFORNIA 92806

(B.)

DC-220
DIAGNOSTIC AND RELIABILITY
SOFTWARE MANUAL
PN 220005

USE TAPES NO. 220011 DIAGNOSTIC
AND NO. 220012 RELIABILITY

```
0001  .MAIN

                ;MOVING HEAD DISK CONTROL DIAGNOSTIC

                ;****** AUTO-RUN AUTO LOAD MODIFIED 3/7/72


                ; 1. ABSTRACT
                ;       THIS PROGRAM IS A HARDWARE DIAGNOSTIC FOR THE
                ;       4046 MOVING HEAD DISK CONTROLLER AND ADAPTER
                ;       LOGIC.  IT IS ASSUMED THAT THE DISK TERMINAL
                ;       IS FUNCTIONING PROPERLY.

                ; 2. REQUIREMENTS
                ;       1. NOVA FAMILY CENTRAL PROCESSOR
                ;       2. MINIMUM OF 4K READ/WRITE MEMORY
                ;       3. 4046 MOVING HEAD DISK CONTROL
                ;       4. 4047,4048,4049 OR 4057
                ;          DISK ADAPTER
                ;       5. 1 TO 4 DISK TERMINALS
                ;       6. TELETYPE AND CONTROL

                ; 3. OPERATING PROCEDURE
                ;       1. LOAD USING THE BINARY LOADER
                ;       2. STARTING ADDRESSES
                ;             SA2= TO IDENTIFY DISK TYPE
                ;                  PROGRAM THEN PROCEEDS TO 400.
                ;             SA4= SET DISK CONTROL ADDRESS TO 33
                ;             SA5= SET DISK CONTROL ADDRESS TO 73
                ;             SA400= START DIAGNOSTIC
                ;       3. THE PROGRAM PRINTS "PASS" FOLLOWING EACH
                ;          COMPLETE PASS THROUGH THE TESTS.
                ;       4. SWITCH SETTINGS
                ;             SW0= FROM ERROR, GO TO NEXT TEST.
                ;             SW1= INHIBIT TELETYPE PRINTING.
                ;             SW2= PRINT FAILURE RATE.
                ;             SW3= RECALIBRATE DURING SCOPE LOOP
                ;             SW4= 1 SEC DELAY IN SCOPE LOOP
                ;             SW5-6= UNIT # FOR RECAL DURING SCOPE LOOP

                ; 4. ERRORS
                ;       WHEN AN ERROR IS DETECTED THE PROGRAM HALTS.
                ;       (AC3) POINTS TO THE LOCATION FOLLOWING THE
                ;       ERROR HALT CALL "EHALT".  CONSULT THE COMMENTS
                ;       AREA OF THE DIAGNOSTIC PROGRAM LISTING FOR
                ;       CLUES AND POSSIBLE CAUSES OF THE FAILURE.
                ;       PUSHING CONTINUE WILL CAUSE THE PROGRAM TO
                ;       PRINT THE (AC3) AND ENTER A SCOPE LOOP.
                ;       SET SW2 TO CAUSE THE ERROR RATE (0-100%) TO BE
                ;       PRINTED.  SET SW0 TO EXIT FROM THE SCOPE LOOP
                ;       AND PROCEED TO THE NEXT TEST.
```

```
;       SOME SCOPE LOOPS WILL REQUIRE A RECALIBRATE
;       TO INITIALIZE THE DISK DRIVE FOLLOWING A FAILURE.
;       SET SWITCH 3 TO INTRODUCE THE RECALIBRATE.  THE
;       UNIT TO BE RECALIBRATED MUST BE SET INTO SWITCHES
;       5 AND 6.

;       TESTS THAT PERFORM A RECALIBRATE HAVE A 2 SEC.
;       DELAY BUILT INTO THE SCOPE LOOP AS PROTECTION
;       FOR THE DISK DRIVE ELECTRONICS.  SET SWITCH 4
;       TO INTRODUCE AN ADDITIONAL 1 SECOND DELAY DURING
;       THE SCOPE LOOP.

;       IN GENERAL EACH SUCCESSIVE TEST ASSUMES ALL
;       PREVIOUS TESTS WORK.  BYPASSING ERRORS
;       CAN RESULT IN CONFUSING SITUATIONS
;       IN THE SETUP OF MORE COMPLEX TESTS.

; 5. DISK PACKS
;       ONLY USE DISK PACKS FORMATTED BY THE DGC DISK
;       PACK FORMATTER PROGRAM.  THE DIAGNOSTIC PROGRAM
;       WILL WRITE OVER MOST OF THE DISK SURFACE.
;       THE FORMAT MODE IS NOT CHECKED.
```

Patch

```
300     44320
  1     24046
  2    125004
  3       404
  4     24320
  5      2401
  6      6227
  7      2401
 10      6233
6224     300
```

Without TY

```
145     17701
146     33551
150    100000
```

(Start at 401)

```
▲ 0003  .MAIN

        000001          .LOC 1

00001 005776            IRET            ;INTERRUPT RETURN
00002 006251            INIT            ;INITIALIZE
00003 002006            JMP @TS         ;GO TO 400
00004 002244            JMP @IS33       ;SET ADDR TO 33
00005 002245            JMP @IS73       ;SET ADDR TO 73
00006 000412 TS:        A1

        000045          .LOC 45
00045 000046            EGGS
00046 000000 EGGS:      0               ;HEN FLAG
00047 000000            0               ;DEVICE CODE THIS RUN
00050 000000            0               ;NOT USED
00051 000000            0               ;# OF PASS THIS RUN
00052 000000            0               ;RETURN ADDRESS

00053 000000 .TU:       0
00054 040000            40000
00055 100000            100000
00056 140000            140000
00057 000001 UNTBIT:    1
00060 000002 KB14:      2
00061 000004 KB13:      4
00062 000010 KB12:      10
00063 000020 KB11:      20
00064 000040 KB10:      40
00065 000100 KB9:       100
00066 000200 KB8:       200
00067 000400 KB7:       400
00070 001000 KB6:       1000
00071 002000 KB5:       2000
00072 004000 KB4:       4000
00073 010000 KB3:       10000
00074 020000 KB2:       20000
        000054 KB1=.TU+1

00075 177775 ZB14:      177775
00076 177773 ZB13:      177773
00077 177767 ZB12:      177767
00100 177757 ZB11:      177757
00101 177737 ZB10:      177737
00102 177677 ZB9:       177677
00103 177577 ZB8:       177577
00104 177377 ZB7:       177377
00105 176777 ZB6:       176777
00106 175777 ZB5:       175777
00107 173777 ZB4:       173777
00110 167777 ZB3:       167777
00111 157777 ZB2:       157777
00112 137777 ZB1:       137777

00113 006231 TRCL:      RECL0
00114 006232            RECL1
00115 006233            RECL2
00116 006234            RECL3
```

```
▲ 0004  .MAIN

000117 000003 C3:      3
000120 000007 C7:      7
000121 000017 C17:     17
000122 000037 C37:     37
000123 000077 C77:     77
000124 000177 C177:    177
000125 000377 C377:    377
000126 000777 C777:    777
000127 001777 C1777:   1777
000130 003777 C3777:   3777
000131 007777 C7777:   7777
000132 017777 C017:    17777
000133 037777 C037:    37777

000134 010421 BIT1:    010421
000135 021042 BIT2:    021042
000136 042104 BIT4:    042104
000137 104210 BIT8:    104210

000140 052525 C2525:   052525
000141 125252 C5252:   125252
000142 123456 RANDOM:  123456
000143 123456 RELRAN:  123456
000144 000000 TEMP:    0
000145 000000 TIME:    0
000146 000000 TIME1:   0
000147 062000 KDOB:    DOR 0,0
000150 000000 DTYPE:   0                ;0=CART, 1=2311, 15=2314
000151 000001 NDSKS:   000001           ;15=UNIT 0, 14=1, 13=2, 12=3
000152 000000 TESTU:   0                ; 0&1 = UNIT #
000153 000000 UNUM:    0                ; 14-15 = UNIT #
000154 000033 CDSK:    33
000155 000000 CYL:     0
000156 000000 HEAD:    0
000157 000000 SECT:    0
000160 006605 BUFF:    PRGEND
000161 160037 MSK1:    160037
000162 060033 DPIO:    060033
000163 177700 MSK2:    177700
000164 000000 ITRCNT:  0
000165 005613 STALL:   .STL
```

```
▲ 0005  .MAIN

00166 000004 C4:      4
00167 000005 C5:      5
00170 000006 C6:      6
00171 000011 C11:     11
00172 000012 C12:     12
00173 000015 C15:     15
00174 000016 C16:     16
00175 000030 C30:     30
00176 000033 C33:     33
00177 000036 C36:     36
00200 000060 C60:     60
00201 000063 C63:     63
00202 000070 C70:     70
00203 000120 C120:    120
00204 000137 C137:    137
00205 000157 C157:    157
00206 000277 C277:    277
00207 000312 C312:    312
00210 000317 C317:    317
00211 000402 C402:    402
00212 000420 C420:    420
00213 177400 C1774:   177400


00214 177400 M400:    -400
00215 001400 C1400:   1400
00216 003600 C3600:   3600
00217 020400 C204H:   20400
00220 074000 C74K:    74000
      000056 C140K=.TU+3
00221 174000 C174K:   174000
      000070 C1000=KB6

00222 024047 BEGIN:   LDA 1,EGGS+1
00223 020176          LDA 0,C33
00224 106414          SUB# 0,1,SZR
00225 002245          JMP @IS73
00226 002244          JMP @IS33
```

```
▲ 0006   .MAIN
  000227 005635 I.WAIT: .WAIT
  000230 005622 I.SSEK: .SSEK
  000231 005704 I.RC0:  .RCL0
  000232 005706 I.RC1:  .RCL1
  000233 005710 I.RC2:  .RCL2
  000234 005711 I.RC3:  .RCL3
  000235 005764 I.IWT:  .IWT
  000236 006000 I.ADSK: .ADSK
  000237 006027 ISET:   .SET
  000240 006032 I.SETP: .SETP
  000241 006035 I.STUP: .SETUP
  000242 006165 I.EHA:  .EHALT
  000243 006065 I.LOO:  .LOOP
  000244 005557 IS33:   .S33
  000245 005560 IS73:   .S73
  000246 006006 I.S:    .SK
  000247 006453 ICRLF:  CRLF
  000250 006310 IMESS:  MESS
  000251 005442 I.INI:  .INI
  000252 005655 ISTB:   STR
  000253 006162 I.EH1:  .EH1
  000254 005722 I.DO:   .DORW
  000255 005217 IRAN:   RAN
  000256 005276 IGEN:   .GEN
  000257 005336 IREAD:  .READ
  000260 005311 IWRT:   .WRITE
  000261 005254 ICHK:   .CHECK
  000262 005354 IDOS:   .DOSEK
  000263 006057 I.LD:   .LDPD

         006241 .DUSR   SETUP=JSR @I.STUP
         006240 .DUSR   SETP1=JSR @I.SETP
         006242 .DUSR   EHALT=JSR @I.EHA
         006243 .DUSR   LOOP=JSR @I.LOO
         000033 .DUSR   DSKP=33
         006253 .DUSR   EHLT=JSR @I.EH1
         006263 .DUSR   LOOPD=JSR @I.LD

         006250 MESSAGE=JSR @IMESS
         006247 PCRLF=JSR @ICRLF
         006227 WAIT=JSR @I.WAIT
         006230 SSEEK=JSR @I.SSEK
         006231 RECL0=JSR @I.RC0
         006232 RECL1=JSR @I.RC1
         006233 RECL2=JSR @I.RC2
         006234 RECL3=JSR @I.RC3
         006235 ITRWT=JSR @I.IWT
         006236 GADSK=JSR @I.ADSK
         006246 SEEK=JSR @I.S
         006251 INIT=JSR @I.INI
         006256 GENDAT=JSR @IGEN
         006257 READ=JSR @IREAD
         006260 WRITE=JSR @IWRT
         006261 CHECK=JSR @ICHK
         006262 DOSEK=JSR @IDOS
         006254 DORW=JSR @I.DO

                .EOT
```

```
0007   .MAIN
       000400          .LOC 400

000400 006252 START:   JSR PISTB
000401 024046          LDA 1,EGGS
000402 125004          MOV 1,1,SZR
000403 000405          JMP A1-2
000404 020150          LDA 0,DTYPE
000405 101005          MOV 0,0,SNR      ;IF PARAMETERS NOT YET SPECIFIED
000406 006251          INIT             ;DO IT NOW.
000407 000403          JMP A1
000410 126620          SUBZR 1,1
000411 044150          STA 1,DTYPE

000412 006241 A1:      SETUP            ;CHECK SELD BUS LINE
000413 063700          SKPDZ 0          ;SKIP IF LINE HIGH
000414 006242          EHALT            ;DSKP HAS SELD GROUNDED.
000415 006243          LOOP

000416 006241 A2:      SETUP            ;CHECK SELB BUS LINE
000417 063500          SKPBZ 0          ;SKIP IF LINE HIGH
000420 006242          EHALT            ;DSKP HAS SELB GROUNDED
000421 006243          LOOP

000422 006241 A3:      SETUP            ;CHECK DISK PACK BUSY
000423 063533          SKPBZ DSKP       ;SKIP IF BUSY 0
000424 006242          EHALT            ;"DP BUSY" STUCK ON
000425 006243          LOOP

000426 006241 A4:      SETUP            ;CHECK DISK PACK DONE
000427 063733          SKPDZ DSKP       ;SKIP IF DONE 0
000430 006242          EHALT            ;"DP DONE" STUCK ON
000431 006243          LOOP

000432 006241 A5:      SETUP            ;CHECK I/0 DATA LINES
000433 060400          DIA 0,0          ;DIA TO DEVICE 0
000434 101004          MOV 0,0,SZR
000435 006242          EHALT            ;GROUNDED DATA LINE(S)
000436 006243          LOOP

000437 006241 A6:      SETUP            ;CHECK CA REGISTER FOR
000440 061433          DIB 0,DSKP       ;ZEROS AFTER "RESET"
000441 101004          MOV 0,0,SZR      ;POSSIBLE FAILURE OF "RESET"
000442 006242          EHALT            ;OR THE REGISTER.
000443 006243          LOOP

000444 006241 A7:      SETUP            ;CHECK DISK ADDRESS REG.
000445 062433          DIC 0,DSKP       ;FOR ZEROS FOLLOWING IORST
000446 101004          MOV 0,0,SZR      ;FAILING REGISTER IC OR
000447 006242          EHALT            ;POSSIBLE FAILURE OF "RESET"
000450 006243          LOOP             ;OR "RESET S".

000451 006241 A8:      SETUP            ;TRY TO LOAD CA WITH
000452 102000          ADC 0,0          ;ALL ONES
000453 062033          DOB 0,DSKP       ;LOAD CA REGISTER
000454 065433          DIB 1,DSKP       ;READ IT BACK
000455 122434          SUBZ# 1,0,SZR    ;CHECK "DP DATOB",
000456 006242          EHALT            ;"DPDATIB", CA REGISTER,
000457 006243          LOOP             ;AND DATA PATH THRU MUX'S.
```

```
00460 006241 A9:    SETUP           ;SEE IF DOB LOADS
00461 102000        ADC 0,0         ;DISK ADDRESS REGISTER
00462 062033        DOB 0,DSKP      ;LOAD CA REG.
00463 066433        DIC 1,DSKP      ;READ DISK ADDR REG.
00464 125004        MOV 1,1,SZR     ;IT SHOULD STILL BE ALL 0'S
00465 006242        EHALT
00466 006243        LOOP

00467 006241 A10:   SETUP           ;SEE IF DOC LOADS THE
00470 102000        ADC 0,0         ;CA REGISTER
00471 063033        DOC 0,DSKP      ;LOAD DISK ADDR REG
00472 065433        DIB 1,DSKP      ;READ CA REGISTER
00473 125004        MOV 1,1,SZR     ;CA REGISTER SHOULD
00474 006242        EHALT           ;REMAIN ALL ZERO

00475 006241 A11:   SETUP           ;SEE IF THE DISK ADDRESS
00476 102000        ADC 0,0         ;REGISTER EXISTS
00477 063033        DOC 0,DSKP      ;LOAD IT WITH ALL 1'S
00500 066433        DIC 1,DSKP      ;READ IT BACK
00501 122414        SUB# 1,0,SZR    ;CHECK REGISTER AND
00502 006242        EHALT           ;DATA PATHS THROUGH
00503 006243        LOOP            ;THE MUX'S

00504 006241 A12:   SETUP           ;SEE IF IORST WILL
00505 102000        ADC 0,0         ;CLEAR THE CA REGISTER
00506 062033        DOB 0,DSKP      ;LOAD IT WITH ALL 1'S
00507 062677        IORST           ;CLEAR IT TO ZEROS (RESET)
00510 065433        DIB 1,DSKP      ;READ IT BACK
00511 125004        MOV 1,1,SZR
00512 006242        EHALT
00513 006243        LOOP

00514 006241 A13:   SETUP           ;SEE IF IORST WILL
00515 102000        ADC 0,0         ;CLEAR DISK ADDRESS
00516 063033        DOC 0,DSKP      ;REGISTER. LOAD ALL 1'S
00517 062677        IORST           ;CLEAR TO ZEROS
00520 066433        DIC 1,DSKP      ;READ BACK
00521 125004        MOV 1,1,SZR     ;(S1,S2,S4,S8,ARE CLEARED
00522 006242        EHALT           ;VIA "RESET" THRU "RESET 5")
00523 006243        LOOP

00524 006241 A14:   SETUP           ;TEST CA REGISTER FOR
00525 102520        SUBZL 0,0       ;SINGLE 1 BITS
00526 062033        DOB 0,DSKP      ;LOAD "CA15"
00527 065433        DIB 1 DSKP      ;READ CA REGISTER
00530 122414        SUB# 1,0,SZR    ;AC0=GOOD
00531 006242        EHALT           ;AC1=BAD
00532 006243        LOOP

00533 006241 A15:   SETUP           ;TEST CA REGISTER FOR
00534 020060        LDA 0,KB14      ;SINGLE 1 BITS
00535 062033        DOB 0,DSKP      ;LOAD "CA14"
00536 065433        DIB 1,DSKP      ;READ CA REGISTER
00537 122414        SUB# 1,0,SZR    ;AC0=GOOD
00540 006242        EHALT           ;AC1=BAD
00541 006243        LOOP
```

```
00542 006241 A16:    SETUP              ;TEST CA REGISTER FOR
00543 020061         LDA 0,KB13         ;SINGLE 1 BITS
00544 062033         DOB 0,DSKP         ;LOAD "CA13"
00545 065433         DIB 1,DSKP         ;READ CA REGISTER
00546 122414         SUB# 1,0,SZR       ;AC0=GOOD
00547 006242         EHALT              ;AC1=BAD
00550 006243         LOOP

00551 006241 A17:    SETUP              ;TEST CA REGISTER FOR
00552 020062         LDA 0,KB12         ;SINGLE 1 BITS
00553 062033         DOB 0,DSKP         ;LOAD "CA12"
00554 065433         DIB 1,DSKP         ;READ CA REGISTER
00555 122414         SUB# 1,0,SZR       ;AC0=GOOD
00556 006242         EHALT              ;AC1=BAD
00557 006243         LOOP

00560 006241 A18:    SETUP              ;TEST CA REGISTER FOR
00561 020063         LDA 0,KB11         ;SINGLE 1 BITS
00562 062033         DOB 0,DSKP         ;LOAD "CA11"
00563 065433         DIB 1,DSKP         ;READ CA REGISTER
00564 122414         SUB# 1,0,SZR       ;AC0=GOOD
00565 006242         EHALT              ;AC1=BAD
00566 006243         LOOP

00567 006241 A19:    SETUP              ;TEST CA REGISTER FOR
00570 020064         LDA 0,KB10         ;SINGLE 1 BITS
00571 062033         DOB 0,DSKP         ;LOAD "CA10"
00572 065433         DIB 1,DSKP         ;READ CA REGISTER
00573 122414         SUB# 1,0,SZR       ;AC0=GOOD
00574 006242         EHALT              ;AC1=BAD
00575 006243         LOOP

00576 006241 A20:    SETUP              ;TEST CA REGISTER FOR
00577 020065         LDA 0,KB9          ;SINGLE 1 BITS
00600 062033         DOB 0,DSKP         ;LOAD "CA9"
00601 065433         DIB 1,DSKP         ;READ CA REGISTER
00602 122414         SUB# 1,0,SZR       ;AC0=GOOD
00603 006242         EHALT              ;AC1=BAD
00604 006243         LOOP

00605 006241 A21:    SETUP              ;TEST CA REGISTER FOR
00606 020066         LDA 0,KB8          ;SINGLE 1 BITS
00607 062033         DOB 0,DSKP         ;LOAD "CA8"
00610 065433         DIB 1,DSKP         ;READ CA REGISTER
00611 122414         SUB# 1,0,SZR       ;AC0=GOOD
00612 006242         EHALT              ;AC1=BAD
00613 006243         LOOP
```

```
00614 006241 A22:    SETUP            ;TEST CA REGISTER FOR
00615 020067         LDA 0,KB7        ;SINGLE 1 BITS
00616 062033         DOR 0,DSKP       ;LOAD "CA7"
00617 065433         DIB 1 DSKP       ;READ CA REGISTER
00620 122414         SUB# 1,0,SZR     ;AC0=GOOD
00621 006242         EHALT            ;AC1=BAD
00622 006243         LOOP

00623 006241 A23:    SETUP            ;TEST CA REGISTER FOR
00624 020033         LDA 0,DSKP       ;SINGLE 1 BITS
00625 061433         DIB 0,DSKP       ;LOAD "CA6"
00626 065433         DIB 1,DSKP       ;READ CA REGISTER
00627 122414         SUB# 1,0,SZR     ;AC0=GOOD
00630 006242         EHALT            ;AC1=BAD

00631 006241 A24:    SETUP            ;TEST CA REGISTER FOR
00632 020071         LDA 0,KB5        ;SINGLE 1 BITS
00633 062033         DOR 0,DSKP       ;LOAD "CA5"
00634 065433         DIB 1,DSKP       ;READ CA REGISTER
00635 122414         SUB# 1,0,SZR     ;AC0=GOOD
00636 006242         EHALT            ;AC1=BAD
00637 006243         LOOP

00640 006241 A25:    SETUP            ;TEST CA REGISTER FOR
00641 020072         LDA 0,KB4        ;SINGLE 1 BITS
00642 062033         DOR 0,DSKP       ;LOAD "CA4"
00643 065433         DIB 1,DSKP       ;READ CA REGISTER
00644 122414         SUB# 1,0,SZR     ;AC0=GOOD
00645 006242         EHALT            ;AC1=BAD
00646 006243         LOOP

00647 006241 A26:    SETUP            ;TEST CA REGISTER FOR
00650 020073         LDA 0,KB3        ;SINGLE 1 BITS
00651 062033         DOR 0,DSKP       ;LOAD "CA3"
00652 065433         DIB 1,DSKP       ;READ CA REGISTER
00653 122414         SUB# 1,0,SZR     ;AC0=GOOD
00654 006242         EHALT            ;AC1=BAD
00655 006243         LOOP

00656 006241 A27:    SETUP            ;TEST CA REGISTER FOR
00657 020074         LDA 0,KB2        ;SINGLE 1 BITS
00660 062033         DOR 0,DSKP       ;LOAD "CA2"
00661 065433         DIB 1,DSKP       ;READ CA REGISTER
00662 122414         SUB# 1,0,SZR     ;AC0=GOOD
00663 006242         EHALT            ;AC1=BAD
00664 006243         LOOP
```

```
00665 006241 A28:    SETUP            ;TEST CA REGISTER FOR
00666 020054         LDA 0,KB1        ;SINGLE 1 BITS
00667 062033         DOB 0,DSKP       ;LOAD "CA1"
00670 065433         DIB 1,DSKP       ;READ CA REGISTER
00671 122414         SUB# 1,0,SZR     ;AC0=GOOD
00672 006242         EHALT            ;AC1=BAD
00673 006243         LOOP

00674 006241 A29:    SETUP            ;TEST CA REGISTER
00675 102620         SUBZR 0,0        ;SINGLE 1 BITS
00676 062033         DOB 0,DSKP       ;LOAD "CA0"
00677 065433         DIB 1,DSKP       ;READ CA REGISTER
00700 122414         SUB# 1,0,SZR     ;AC0=GOOD
00701 006242         EHALT            ;AC1=BAD
00702 006243         LOOP

00703 006241 A30:    SETUP            ;TEST CA REGISTER FOR
00704 102120         ADCZL 0,0        ;SINGLE 0 BITS
00705 062033         DOB 0,DSKP       ;LOAD 177776
00706 065433         DIB 1,DSKP       ;READ CA REGISTER
00707 122414         SUB# 1,0,SZR     ;AC0=GOOD
00710 006242         EHALT            ;AC1=BAD
00711 006243         LOOP

00712 006241 A31:    SETUP            ;TEST CA REGISTER FOR
00713 020075         LDA 0,ZB14       ;SINGLE 0 BITS
00714 062033         DOB 0,DSKP       ;LOAD 177775
00715 065433         DIB 1,DSKP       ;READ CA REGISTER
00716 122414         SUB# 1,0,SZR     ;AC0=GOOD
00717 006242         EHALT            ;AC1=BAD
00720 006243         LOOP

00721 006241 A32:    SETUP            ;TEST CA REGISTER FOR
00722 020076         LDA 0,ZB13       ;SINGLE 0 BITS
00723 062033         DOB 0,DSKP       ;LOAD 177773
00724 065433         DIB 1,DSKP       ;READ CA REGISTER
00725 122414         SUB# 1,0,SZR     ;AC0=GOOD
00726 006242         EHALT            ;AC1=BAD
00727 006243         LOOP

00730 006241 A33:    SETUP            ;TEST CA REGISTER FOR
00731 020077         LDA 0,ZB12       ;SINGLE 0 BITS
00732 062033         DOB 0,DSKP       ;LOAD 177767
00733 065433         DIB 1,DSKP       ;READ CA REGISTER
00734 122414         SUB# 1,0,SZR     ;AC0=GOOD
00735 006242         EHALT            ;AC1=BAD
00736 006243         LOOP
```

```
00737 006241 A34:   SETUP           ;TEST CA REGISTER FOR
00740 020100        LDA 0,ZB11      ;SINGLE 0 BITS
00741 062033        DOB 0,DSKP      ;LOAD 177757
00742 065433        DIB 1,DSKP      ;READ CA REGISTER
00743 122414        SUB# 1,0,SZR    ;AC0=GOOD
00744 006242        EHALT           ;AC1=BAD
00745 006243        LOOP

00746 006241 A35:   SETUP           ;TEST CA REGISTER FOR
00747 020101        LDA 0,ZB10      ;SINGLE ZERO BITS
00750 062033        DOB 0,DSKP      ;LOAD 177737
00751 065433        DIB 1,DSKP      ;READ CA REGISTER
00752 122414        SUB# 1,0,SZR    ;AC0=GOOD
00753 006242        EHALT           ;AC1=BAD
00754 006243        LOOP

00755 006241 A36:   SETUP           ;TEST CA REGISTER FOR
00756 020102        LDA 0,ZB9       ;SINGLE ZERO BITS
00757 062033        DOB 0,DSKP      ;LOAD 177677
00760 065433        DIB 1,DSKP      ;READ CA REGISTER
00761 122414        SUB# 1,0,SZR    ;AC0=GOOD
00762 006242        EHALT           ;AC1=BAD
00763 006243        LOOP

00764 006241 A37:   SETUP           ;TEST CA REGISTER FOR
00765 020103        LDA 0,ZB8       ;SINGLE 0 BITS
00766 062033        DOB 0,DSKP      ;LOAD 177577
00767 065433        DIB 1,DSKP      ;READ CA REGISTER
00770 122414        SUB# 1,0,SZR    ;AC0=GOOD
00771 006242        EHALT           ;AC1=BAD
00772 006243        LOOP

00773 006241 A38:   SETUP           ;TEST CA REGISTER FOR
00774 020104        LDA 0,ZB7       ;SINGLE 0 BITS
00775 062033        DOB 0,DSKP      ;LOAD 177377
00776 065433        DIB 1,DSKP      ;READ CA REGISTER
00777 122414        SUB# 1,0,SZR    ;AC0=GOOD
01000 006242        EHALT           ;AC1=BAD
01001 006243        LOOP

01002 006241 A39:   SETUP           ;TEST CA REGISTER FOR
01003 020105        LDA 0,ZB6       ;SINGLE ZERO BITS
01004 062033        DOB 0,DSKP      ;LOAD 176777
01005 065433        DIB 1,DSKP      ;READ CA REGISTER
01006 122414        SUB# 1,0,SZR    ;AC0=GOOD
01007 006242        EHALT           ;AC1=BAD
01010 006243        LOOP
```

```
01011 006241 A40:    SETUP            ;TEST CA REGISTER FOR
01012 020106         LDA 0,ZB5        ;SINGLE ZERO BITS
01013 062033         DOB 0,DSKP       ;LOAD 175777
01014 065433         DIB 1,DSKP       ;READ CA REGISTER
01015 122414         SUB# 1,0,SZR     ;AC0=GOOD
01016 006242         EHALT            ;AC1=BAD
01017 006243         LOOP

01020 006241 A41:    SETUP            ;TEST CA REGISTER FOR
01021 020107         LDA 0,ZB4        ;SINGLE 0 BITS
01022 062033         DOB 0,DSKP       ;LOAD 173777
01023 065433         DIB 1,DSKP       ;READ CA REGISTER
01024 122414         SUB# 1,0,SZR     ;AC0=GOOD
01025 006242         EHALT            ;AC1=BAD
01026 006243         LOOP

01027 006241 A42:    SETUP            ;TEST CA REGISTER FOR
01030 020110         LDA 0,ZB3        ;SINGLE 0 BITS
01031 062033         DOB 0,DSKP       ;LOAD 167777
01032 065433         DIB 1,DSKP       ;READ CA REGISTER
01033 122414         SUB# 1,0,SZR     ;AC0=GOOD
01034 006242         EHALT            ;AC1=BAD
01035 006243         LOOP

01036 006241 A43:    SETUP            ;TEST CA REGISTER FOR
01037 020111         LDA 0,ZB2        ;SINGLE ZERO BITS
01040 062033         DOB 0,DSKP       ;LOAD 157777
01041 065433         DIB 1,DSKP       ;READ CA REGISTER
01042 122414         SUB# 1,0,SZR     ;AC0=GOOD
01043 006242         EHALT            ;AC1=BAD
01044 006243         LOOP

01045 006241 A44:    SETUP            ;TEST CA REGISTER FOR
01046 020112         LDA 0,ZB1        ;SINGLE 0 BIT
01047 062033         DOB 0,DSKP       ;LOAD 137777
01050 065433         DIB 1,DSKP       ;READ CA REGISTER
01051 122414         SUB# 1,0,SZR     ;AC0=GOOD
01052 006242         EHALT            ;AC1=BAD
01053 006243         LOOP

01054 006241 A45:    SETUP            ;TEST CA REGISTER FOR
01055 102220         ADCZR 0,0        ;SINGLE ZERO BIT
01056 062033         DOB 0,DSKP       ;LOAD 077777
01057 065433         DIB 1,DSKP       ;READ CA REGISTER
01060 122414         SUB# 1,0,SZR     ;AC0=GOOD
01061 006242         EHALT            ;AC1=BAD
01062 006243         LOOP
```

```
001063 006241 A46:    SETUP           ;TEST DISK ADDRESS REGISTER
001064 102520         SUBZL 0,0       ;FOR SINGLE 1 BIT
001065 063033         DOC 0,DSKP      ;LOAD "SC1"
001066 066433         DIC 1,DSKP      ;READ BACK
001067 122414         SUB# 1,0,SZR    ;AC0=GOOD
001070 006242         EHALT           ;AC1=BAD
001071 006243         LOOP

001072 006241 A47:    SETUP           ;TEST DISK ADDRESS REGISTER
001073 020060         LDA 0,KB14      ;FOR SINGLE 1 BITS
001074 063033         DOC 0,DSKP      ;LOAD "SC2"
001075 066433         DIC 1,DSKP      ;READ BACK
001076 122414         SUB# 1,0,SZR    ;AC0=GOOD
001077 006242         EHALT           ;AC1=BAD
001100 006243         LOOP

001101 006241 A48:    SETUP           ;TEST DISK ADDRESS REGISTER
001102 020061         LDA 0,KB13      ;FOR SINGLE 1 BITS
001103 063033         DOC 0,DSKP      ;LOAD "SC4"
001104 066433         DIC 1,DSKP      ;READ BACK
001105 122414         SUB# 1,0,SZR    ;AC0=GOOD
001106 006242         EHALT           ;AC1=BAD
001107 006243         LOOP

001110 006241 A49:    SETUP           ;TEST DISK ADDRESS REGISTER
001111 020062         LDA 0,KB12      ;FOR SINGLE 1 BITS
001112 063033         DOC 0,DSKP      ;LOAD "SC8"
001113 066433         DIC 1,DSKP      ;READ BACK
001114 122414         SUB# 1,0,SZR    ;AC0=GOOD
001115 006242         EHALT           ;AC1=BAD
001116 006243         LOOP

001117 006241 A50:    SETUP           ;TEST DISK ADDRESS REGISTER
001120 020063         LDA 0,KB11      ;FOR SINGLE 1 BITS
001121 063033         DOC 0,DSKP      ;LOAD "S1"
001122 066433         DIC 1,DSKP      ;READ BACK
001123 122414         SUB# 1,0,SZR    ;AC0=GOOD
001124 006242         EHALT           ;AC1=BAD
001125 006243         LOOP

001126 006241 A51:    SETUP           ;TEST DISK ADDRESS REGISTER
001127 020064         LDA 0,KB10      ;FOR SINGLE 1 BITS
001130 063033         DOC 0,DSKP      ;LOAD "S2"
001131 066433         DIC 1,DSKP      ;READ BACK
001132 122414         SUB# 1,0,SZR    ;AC0=GOOD
001133 006242         EHALT           ;AC1=BAD
001134 006243         LOOP
```

```
01135 006241 A52:    SETUP            ;TEST DISK ADDRESS REGISTER
01136 020065         LDA 0,KB9        ;FOR SINGLE 1 BITS
01137 063033         DOC 0,DSKP       ;LOAD "S4"
01140 066433         DIC 1,DSKP       ;READ IT BACK
01141 122414         SUB# 1,0,SZR     ;AC0=GOOD
01142 006242         EHALT            ;AC1=BAD
01143 006243         LOOP

01144 006241 A53:    SETUP            ;TEST DISK ADDRESS REGISTER
01145 020066         LDA 0,KB8        ;FOR SINGLE 1 BITS
01146 063033         DOC 0,DSKP       ;LOAD "S8"
01147 066433         DIC 1,DSKP       ;READ IT BACK
01150 122414         SUB# 1,0,SZR     ;AC0=GOOD
01151 006242         EHALT            ;AC1=BAD
01152 006243         LOOP

01153 006241 A54:    SETUP            ;TEST DISK ADDRESS REGISTER
01154 020067         LDA 0,KB7        ;FOR SINGLE 1 BITS
01155 063033         DOC 0,DSKP       ;LOAD "HD1"
01156 066433         DIC 1,DSKP       ;READ IT BACK
01157 122414         SUB# 1,0,SZR     ;AC0=GOOD
01160 006242         EHALT            ;AC1=BAD
01161 006243         LOOP

01162 006241 A55:    SETUP            ;TEST DISK ADDRESS REGISTER
01163 020070         LDA 0,KB6        ;FOR SINGLE 1 BITS
01164 063033         DOC 0,DSKP       ;LOAD "HD2"
01165 066433         DIC 1,DSKP       ;READ BACK
01166 122414         SUB# 1,0,SZR     ;AC0=GOOD
01167 006242         EHALT            ;AC1=BAD
01170 006243         LOOP

01171 006241 A56:    SETUP            ;TEST DISK ADDRESS REGISTER
01172 020071         LDA 0,KB5        ;FOR SINGLE 1 BITS
01173 063033         DOC 0,DSKP       ;LOAD "HD4"
01174 066433         DIC 1,DSKP       ;READ IT BACK
01175 122414         SUB# 1,0,SZR     ;AC0=GOOD
01176 006242         EHALT            ;AC1=BAD
01177 006243         LOOP

01200 006241 A57:    SETUP            ;TEST DISK ADDRESS REGISTER
01201 020072         LDA 0,KB4        ;FOR SINGLE 1 BITS
01202 063033         DOC 0,DSKP       ;LOAD "HD8"
01203 066433         DIC 1,DSKP       ;READ IT BACK
01204 122414         SUB# 1,0,SZR     ;AC0=GOOD
01205 006242         EHALT            ;AC1=BAD
01206 006243         LOOP
```

```
01207 006241 A58:    SETUP            ;TEST DISK ADDRESS REGISTER
01210 020073         LDA 0,KB3        ;FOR SINGLE 1 BITS
01211 063033         DOC 0,DSKP       ;LOAD "HD16"
01212 066433         DIC 1,DSKP       ;READ IT BACK
01213 122414         SUB# 1,0,SZR     ;AC0=GOOD
01214 006242         EHALT            ;AC1=BAD
01215 006243         LOOP

01216 006241 A59:    SETUP            ;TEST DISK ADDRESS REGISTER
01217 020074         LDA 0,KB2        ;FOR SINGLE 1 BITS
01220 063033         DOC 0,DSKP       ;LOAD "FORMAT"
01221 066433         DIC 1,DSKP       ;READ IT BACK
01222 122414         SUB# 1,0,SZR     ;AC0=GOOD
01223 006242         EHALT            ;AC1=BAD
01224 006243         LOOP

01225 006241 A60:    SETUP            ;TEST DISK ADDRESS REGISTER
01226 020054         LDA 0,KB1        ;FOR SINGLE 1 BIT
01227 063033         DOC 0,DSKP       ;LOAD "D1"
01230 066433         DIC 1,DSKP       ;READ IT BACK
01231 122414         SUB# 1,0,SZR     ;AC0=GOOD
01232 006242         EHALT            ;AC1=BAD
01233 006243         LOOP

01234 006241 A61:    SETUP            ;TEST DISK ADDRESS REGISTER
01235 102620         SUBZR 0,0        ;FOR SINGLE 1 BITS
01236 063033         DOC 0,DSKP       ;LOAD "D0"
01237 066433         DIC 1,DSKP       ;READ IT BACK
01240 122414         SUB# 1,0,SZR     ;AC0=GOOD
01241 006242         EHALT            ;AC1=BAD
01242 006243         LOOP

01243 006241 A62:    SETUP            ;TEST DISK ADDRESS REGISTER
01244 102120         ADCZL 0,0        ;FOR SINGLE 0 BITS
01245 063033         DOC 0,DSKP       ;"SC1"=0
01246 066433         DIC 1,DSKP       ;READ THE REGISTER
01247 122414         SUB# 1,0,SZR     ;AC0=GOOD
01250 006242         EHALT            ;AC1=BAD
01251 006243         LOOP

01252 006241 A63:    SETUP            ;TEST DISK ADDRESS REGISTER
01253 020075         LDA 0,ZB14       ;FOR SINGLE 0 BITS
01254 063033         DOC 0,DSKP       ;"SC2" = 0
01255 066433         DIC 1,DSKP       ;READ REGISTER
01256 122414         SUB# 1,0,SZR     ;AC0=GOOD
01257 006242         EHALT            ;AC1=BAD
01260 006243         LOOP
```

```
01261 006241 A64:    SETUP            ;TEST DISK ADDRESS REGISTER
01262 020076         LDA 0,ZB13       ;FOR SINGLE 0 BITS
01263 063033         DOC 0,DSKP       ;"SC4" = 0
01264 066433         DIC 1,DSKP       ;READ REGISTER
01265 122414         SUB# 1,0,SZR     ;AC0=GOOD
01266 006242         EHALT            ;AC1=BAD
01267 006243         LOOP

01270 006241 A65:    SETUP            ;TEST DISK ADDRESS REGISTER
01271 020077         LDA 0,ZB12       ;FOR SINGLE 0 BITS
01272 063033         DOC 0,DSKP       ;"SC8" = 0
01273 066433         DIC 1,DSKP       ;READ REGISTER
01274 122414         SUB# 1,0,SZR     ;AC0=GOOD
01275 006242         EHALT            ;AC1=BAD
01276 006243         LOOP

01277 006241 A66:    SETUP            ;TEST DISK ADDRESS REGISTER
01300 020100         LDA 0,ZB11       ;FOR SINGLE 0 BITS
01301 063033         DOC 0,DSKP       ;"S1" = 0
01302 066433         DIC 1,DSKP       ;READ REGISTER
01303 122414         SUB# 1,0,SZR     ;AC0=GOOD
01304 006242         EHALT            ;AC1=BAD
01305 006243         LOOP

01306 006241 A67:    SETUP            ;TEST DISK ADDRESS REGISTER
01307 020101         LDA 0,ZB10       ;FOR SINGLE ZERO BITS
01310 063033         DOC 0,DSKP       ;"S2" = 0
01311 066433         DIC 1,DSKP       ;READ IT BACK
01312 122414         SUB# 1,0,SZR     ;AC0=GOOD
01313 006242         EHALT            ;AC1=BAD
01314 006243         LOOP

01315 006241 A68:    SETUP            ;TEST DISK ADDRESS REGISTER
01316 020102         LDA 0,ZB9        ;FOR SINGLE ZERO BIT
01317 063033         DOC 0,DSKP       ;"S4" = 0
01320 066433         DIC 1,DSKP       ;READ IT BACK
01321 122414         SUB# 1,0,SZR     ;AC0=GOOD
01322 006242         EHALT            ;AC1=BAD
01323 006243         LOOP

01324 006241 A69:    SETUP            ;TEST DISK ADDRESS REGISTER
01325 020103         LDA 0,ZB8        ;FOR SINGLE ZERO BITS
01326 063033         DOC 0,DSKP       ;"S8" = 0
01327 066433         DIC 1,DSKP       ;READ IT BACK
01330 122414         SUB# 1,0,SZR     ;AC0=GOOD
01331 006242         EHALT            ;AC1=BAD
01332 006243         LOOP
```

```
▲ 0018  .MAIN

 01333 000241 A70:   SETUP           ;TEST DISK ADDRESS REGISTER
 01334 020104        LDA 0,7B7       ;FOR SINGLE 0 BIT
 01335 063033        DOC 0,DSKP      ;"HD1" = 0
 01336 066433        DIC 1,DSKP      ;READ IT BACK
 01337 122414        SUB# 1,0,SZR    ;AC0=GOOD
 01340 000242        EHALT           ;AC1=BAD
 01341 000243        LOOP

 01342 000241 A71:   SETUP           ;TEST DISK ADDRESS REGISTER
 01343 020105        LDA 0,7B6       ;FOR SINGLE 0 BIT
 01344 063033        DOC 0,DSKP      ;"HD2" = 0
 01345 066433        DIC 1,DSKP      ;READ IT BACK
 01346 122414        SUB# 1,0,SZR    ;AC0=GOOD
 01347 000242        EHALT           ;AC1=BAD
 01350 000243        LOOP

 01351 000241 A72:   SETUP           ;TEST DISK ADDRESS REGISTER
 01352 020106        LDA 0,7B5       ;FOR SINGLE ZERO BITS
 01353 063033        DOC 0,DSKP      ;"HD4" = 0
 01354 066433        DIC 1,DSKP      ;READ IT BACK
 01355 122414        SUB# 1,0,SZR    ;AC0=GOOD
 01356 000242        EHALT           ;AC1=BAD
 01357 000243        LOOP

 01360 000241 A73:   SETUP           ;TEST DISK ADDRESS REGISTER
 01361 020107        LDA 0,7B4       ;FOR SINGLE 0 BIT
 01362 063033        DOC 0,DSKP      ;"HD8" = 0
 01363 066433        DIC 1,DSKP      ;READ IT BACK
 01364 122414        SUB# 1,0,SZR    ;AC0=GOOD
 01365 000242        EHALT           ;AC1=BAD
 01366 000243        LOOP

 01367 000241 A74:   SETUP           ;TEST DISK ADDRESS REGISTER
 01370 020110        LDA 0,7B3       ;FOR SINGLE 0 BIT
 01371 063033        DOC 0,DSKP      ;"HD16" = 0
 01372 066433        DIC 1,DSKP      ;READ IT BACK
 01373 122414        SUB# 1,0,SZR    ;AC0=GOOD
 01374 000242        EHALT           ;AC1=BAD
 01375 000243        LOOP

 01376 000241 A75:   SETUP           ;TEST DISK ADDRESS REGISTER
 01377 020111        LDA 0,7B2       ;FOR SINGLE ZERO BITS
 01400 063033        DOC 0,DSKP      ;"FORMAT" = 0
 01401 066433        DIC 1,DSKP      ;READ IT BACK
 01402 122414        SUB# 1,0,SZR    ;AC0=GOOD
 01403 000242        EHALT           ;AC1=BAD
 01404 000243        LOOP

                     .EOT
```

```
0019  .MAIN

01405 006241 A76:    SETUP            ;TEST DISK ADDRESS REGISTER
01406 020112         LDA 0,ZB1        ;FOR SINGLE 0 BIT
01407 063033         DOC 0,DSKP       ;"D1" = 0
01410 066433         DIC 1,DSKP       ;READ IT BACK
01411 122414         SUB# 1,0,SZR     ;AC0=GOOD
01412 006242         EHALT            ;AC1=BAD WORD
01413 006243         LOOP

01414 006241 A77:    SETUP            ;TEST DISK ADDRESS REGISTER
01415 102220         ADCZR 0,0        ;FOR SINGLE ZERO BIT
01416 063033         DOC 0,DSKP       ;"D0" = 0
01417 066433         DIC 1,DSKP       ;READ IT BACK
01420 122414         SUB# 1,0,SZR     ;AC0=GOOD
01421 006242         EHALT            ;AC1=BAD
01422 006243         LOOP

01423 102400 A78:    SUB 0,0          ;CHECK CA REGISTER, ALL
01424 006237         JSR @ISET        ;POSSIBLE PATTERNS
01425 062033         DOB 0,DSKP       ;LOAD CA
01426 065433         DIB 1,DSKP       ;READ IT BACK
01427 122414         SUB# 1,0,SZR     ;AC0=GOOD
01430 006242         EHALT            ;AC1=BAD
01431 006243         LOOP             ;DO IT ONLY ONCE FOR EACH PAT
01432 101404         INC 0,0,SZR      ;NEXT PATTERN
01433 000771         JMP .-7

01434 006237 A79:    JSR @ISET        ;CHECK DISK ADDRESS REGISTER
01435 063033         DOC 0,DSKP       ;ALL POSSIBLE PATTERNS
01436 066433         DIC 1,DSKP       ;LOAD/READ BACK
01437 122414         SUB# 1,0,SZR     ;AC0=GOOD
01440 006242         EHALT            ;AC1=BAD
01441 006243         LOOP
01442 101404         INC 0,0,SZR      ;NEXT PATTERN
01443 000771         JMP .-7

01444 006241 A80:    SETUP            ;SPECIAL TEST FOR 8291
01445 102400         SUB 0,0          ;IC PACKS.  SET RIGHTMOST
01446 030134         LDA 2,BIT1       ;BIT IN EACH PACK WITH
01447 072033         DOB 2,DSKP       ;3 SUCCESSIVE LOADS, THEN
01450 072033         DOB 2,DSKP       ;LOAD ZERO.  MARGINAL
01451 072033         DOB 2,DSKP       ;PACKS WILL HOLD THE "1".
01452 062033         DOB 0,DSKP
01453 065433         DIB 1,DSKP       ;TESTING CA REGISTER
01454 125004         MOV 1,1,SZR      ;AC0=GOOD
01455 006242         EHALT            ;AC1=BAD
01456 006243         LOOP
```

```
001457 006241 A81:    SETUP             ;SPECIAL TEST FOR 8291
001460 102400         SUB 0,0           ;IC PACKS.  SET POSITION 2
001461 030135         LDA 2,BIT2        ;(PIN 9) IN EACH PACK WITH
001462 072033         DOB 2,DSKP        ;3 SUCCESSIVE LOADS, THEN
001463 072033         DOB 2,DSKP        ;LOAD ZEROS.  MARGINAL
001464 072033         DOB 2,DSKP        ;PACKS WILL HOLD A "1"
001465 062033         DOB 0,DSKP
001466 065433         DIB 1,DSKP        ;TESTING CA REGISTER
001467 122414         SUB# 1,0,SZR      ;AC0=GOOD
001470 006242         EHALT             ;AC1=BAD
001471 006243         LOOP

001472 006241 A82:    SETUP             ;SPECIAL TEST FOR 8291
001473 102400         SUB 0,0           ;IC PACKS.  SET POSITION 4
001474 030136         LDA 2,BIT4        ;(PIN 2) IN EACH PACK WITH
001475 072033         DOB 2,DSKP        ;3 SUCCESSIVE LOADS, THEN
001476 072033         DOB 2,DSKP        ;LOAD ZEROS.  MARGINAL
001477 072033         DOB 2,DSKP        ;PACKS WILL HOLD A "1".
001500 062033         DOB 0,DSKP
001501 065433         DIB 1,DSKP        ;TESTING CA REGISTER
001502 122414         SUB# 1,0,SZR      ;AC0=GOOD
001503 006242         EHALT             ;AC1=BAD
001504 006243         LOOP

001505 006241 A83:    SETUP             ;SPECIAL TEST FOR 8291
001506 102400         SUB 0,0           ;IC PACKS.  SET POSITION 8
001507 030137         LDA 2,BIT8        ;(PIN 12) IN EACH PACK WITH
001510 072033         DOB 2,DSKP        ;3 SUCCESSIVE LOADS.  THEN
001511 072033         DOB 2,DSKP        ;LOAD ZEROS.  MARGINAL
001512 072033         DOB 2,DSKP        ;PACKS WILL HOLD A "1"
001513 062033         DOB 0,DSKP
001514 065433         DIB 1,DSKP        ;TESTING CA REGISTER
001515 122414         SUB# 1,0,SZR      ;AC0=GOOD
001516 006242         EHALT             ;AC1=BAD
001517 006243         LOOP

001520 006241 A84:    SETUP             ;SPECIAL TEST FOR 8291
001521 102400         SUB 0,0           ;IC PACKS.  SET POSITION 1
001522 030134         LDA 2,BIT1        ;(PIN 5) OF EACH PACK WITH
001523 073033         DOC 2,DSKP        ;3 SUCCESSIVE LOADS, THEN
001524 073033         DOC 2,DSKP        ;LOAD ZEROS.  MARGINAL
001525 073033         DOC 2,DSKP        ;PACKS WILL HOLD A "1".
001526 063033         DOC 0,DSKP
001527 066433         DIC 1,DSKP        ;TESTING DISK ADDR REG.
001530 122414         SUB# 1,0,SZR      ;AC0=GOOD
001531 006242         EHALT             ;AC1=BAD
001532 006243         LOOP
```

```
001533 006241 A85:    SETUP             ;SPECIAL TEST FOR 8291
001534 102400         SUB 0,0           ;IC PACKS.  SET POSITION 2
001535 030135         LDA 2,BIT2        ;(PIN 9) OF EACH PACK WITH
001536 073033         DOC 2,DSKP        ;3 SUCCESSIVE LOADS, THEN
001537 073033         DOC 2,DSKP        ;LOAD ZEROS.  MARGINAL
001540 073033         DOC 2,DSKP        ;PACKS WILL RETAIN A "1".
001541 063033         DOC 0,DSKP
001542 066433         DIC 1,DSKP        ;TESTING DISK ADDR REG.
001543 122414         SUB# 1,0,SZR      ;AC0=GOOD
001544 006242         EHALT             ;AC1=BAD
001545 006243         LOOP

001546 006241 A86:    SETUP             ;SPECIAL TEST FOR 8291
001547 102400         SUB 0,0           ;IC PACKS.  SET POSITION 4
001550 030136         LDA 2,BIT4        ;(PIN 2) OF EACH PACK WITH
001551 073033         DOC 2,DSKP        ;3 SUCCESSIVE LOADS, THEN
001552 073033         DOC 2,DSKP        ;LOAD ZEROS.  MARGINAL
001553 073033         DOC 2,DSKP        ;PACKS WILL RETAIN A "1".
001554 063033         DOC 0,DSKP.
001555 066433         DIC 1,DSKP        ;TESTING DISK ADDR REG.
001556 122414         SUB# 1,0,SZR      ;AC0=GOOD
001557 006242         EHALT             ;AC1=BAD
001560 006243         LOOP

001561 006241 A87:    SETUP             ;SPECIAL TEST FOR 8291
001562 102400         SUB 0,0           ;IC PACKS.  SET POSITION 8
001563 030137         LDA 2,BIT8        ;(PIN 12) OF EACH PACK WITH
001564 073033         DOC 2,DSKP        ;3 SUCCESSIVE LOADS, THEN
001565 073033         DOC 2,DSKP        ;LOAD ALL ZEROS.  MARGINAL
001566 073033         DOC 2,DSKP        ;PACKS WILL RETAIN A "1".
001567 063033         DOC 0,DSKP
001570 066433         DIC 1,DSKP        ;TESTING DISK ADDR REG.
001571 122414         SUB# 1,0,SZR      ;AC0=GOOD
001572 006242         EHALT             ;AC1=BAD
001573 006243         LOOP

001574 020147 B1:     LDA 0,KDOB        ;CHECK FOR ILLEGAL
001575 040403         STA 0,B1.1        ;DEVICE SELECT BY
001576 024140         LDA 1,C2525       ;DISK PACK CONTROL
001577 006240         SETP1             ;PERFORM "DOB" TO EVERY
001600 060000 B1.1:   DOB 1,0           ;OTHER DEVICE ADDRESS
001601 061433         DIB 0,DSKP        ;AND CHECK DSKP
001602 122415         SUB# 1,0,SNR      ;EACH TIME TO SEE
001603 006242         EHALT             ;IT RECOGNIZED THE "DOB".
001604 006243         LOOP
001605 010773 B1.2:   ISZ B1.1          ;CODE FROM HERE ON IS
001606 020772         LDA 0,B1.1        ;FOR INCREMENTING TO
001607 030123         LDA 2,C77         ;NEXT DEVICE ADDRESS
001610 143405         AND 2,0,SNR
001611 000405         JMP B2
001612 030154         LDA 2,COSK
001613 142415         SUB# 2,0,SNR
001614 000771         JMP B1.2
001615 000762         JMP B1.1-1
```

```
01616 006241 B2:     SETUP           ;SEE IF (S) PULSE WILL SET
01617 020070         LDA 0,C1000     ;"DP BUSY".  START A
01620 061133         DOAS 0,DSKP     ;SEEK
01621 063433         SKPBN DSKP      ;SKIP IF "DP BUSY" = 1
01622 006253         EHLT            ;CHECK "DP START", "DP BUSY",
01623 006263         LOOPD           ;"SELB" OC GATE.

01624 006241 B3:     SETUP           ;SEE IF (C) PULSE WILL CLEAR
01625 020070         LDA 0,C1000     ;"DP BUSY".  START A SEEK.
01626 061133         DOAS 0,DSKP     ; THEN CLEAR
01627 060233         NIOC DSKP
01630 063533         SKPBZ DSKP      ;CHECK "CLEAR"
01631 006253         EHLT
01632 006263         LOOPD

01633 006241 B4:     SETUP           ;SEE IF IORST WILL CLEAR
01634 020070         LDA 0,C1000     ;"DP BUSY".  START A SEEK.
01635 061133         DOAS 0,DSKP     ; THEN CLEAR IT
01636 062677         IORST
01637 063533         SKPBZ DSKP      ;CHECK "RESET", "CLEAR"
01640 006253         EHLT
01641 006263         LOOPD

01642 006241 B5:     SETUP           ;CHECK TO INSURE THAT
01643 064433         DIA 1,DSKP      ;ALL SEEKING FF'S ARE
01644 020216         LDA 0,C3600     ;CLEARED BY IORST
01645 107404         AND 0,1,SZR     ;AC1=BAD SEEKING
01646 006242         EHALT           ;STATUS FROM DIA.
01647 006243         LOOP            ;CHECK DATA THRU MUX

01650 006241 B6:     SETUP           ;CHECK FOR PROPER
01651 102000         ADC 0,0         ;INPUT OF STATUS THRU
01652 062033         DOB 0,DSKP      ;THE MULTIPLEXORS TO
01653 063033         DOC 0,DSKP      ;THE DATA BUSS OC GATES
01654 064433         DIA 1,DSKP
01655 020216         LDA 0,C3600
01656 107414         AND# 0,1,SZR
01657 006242         EHALT           ;SEEKING STATUS ON
01660 006243         LOOP            ;ALL SHOULD BE ZERO. (AC1)=STATUS

01661 006241 B7:     SETUP           ;TRY TO SET "SEEKING 0"
01662 020070         LDA 0,C1000     ;VIA (S) PULSE
01663 061133         DOAS 0,DSKP
01664 064433         DIA 1,DSKP      ;"DP START" SETS "START"
01665 020071         LDA 0,KB5       ;CHECK "ADAPTER SEL", "SEEK"
01666 123415         AND# 1,0,SNR    ;AND "START SEEK"
01667 006253         EHLT
01670 006263         LOOPD
```

```
001671 006241 BB:     SETUP             ;TRY TO SET "SEEKING 0"
001672 020070         LDA 0,C1000       ;WITH A (P) PULSE
001673 061333         DOAP 0,DSKP
001674 024071         LDA 1,KB5         ;CHECK "DP IOP"
001675 060433         DIA 0,DSKP        ;SET LINE TO "START"
001676 123415         AND# 1,0,SNR
001677 006253         EHLT
001700 006263         LOOPD

001701 006241 B9:     SETUP             ;CHECK UNIT SELECTION
001702 020070         LDA 0,C1000       ;START UNIT 0 SEEKING
001703 061333         DOAP 0,DSKP       ;AND VERIFY THAT NO
001704 070433         DIA 2,DSKP        ;OTHER UNIT SEEKS.
001705 020071         LDA 0,KB5         ;AC2=STATUS DURING SEEK
001706 024216         LDA 1,C3600       ;AC1=BAD SEEKING STATUS
001707 147400         AND 2,1           ;AC0=GOOD
001710 122414         SUB# 1,0,SZR      ;CHECK UNIT # DECODER
001711 006253         EHLT
001712 006263         LOOPD

001713 006241 B10:    SETUP             ;CHECK UNIT SELECTION 1
001714 020054         LDA 0,KB1         ;START UNIT 1 SEEKING
001715 063033         DOC 0,DSKP        ;AND VERIFY THAT NO
001716 020070         LDA 0,C1000       ;OTHER UNIT SEEKS.
001717 061333         DOAP 0,DSKP       ;AC2=STATUS DURING SEEK
001720 070433         DIA 2,DSKP        ;AC1=BAD SEEKING STATUS
001721 024216         LDA 1,C3600       ;AC0=GOOD
001722 020072         LDA 0,KB6         ;CHECK UNIT # DECODER,
001723 147400         AND 2,1           ;"UNIT 1", AND "SEEKING 1"
001724 122414         SUB# 1,0,SZR
001725 006253         EHLT
001726 006263         LOOPD

001727 006241 B11:    SETUP             ;CHECK UNIT SELECTION
001730 102620         SUBZR 0,0         ;START UNIT 2 SEEKING AND
001731 063033         DOC 0,DSKP        ;VERIFY THAT NO OTHER
001732 020070         LDA 0,C1000       ;UNIT SEEKS
001733 061333         DOAP 0,DSKP       ;AC2=STATUS DURING SEEK
001734 070433         DIA 2,DSKP        ;AC1=BAD SEEKING STATUS
001735 024216         LDA 1,C3600       ;AC0=GOOD
001736 020067         LDA 0,KB7         ;CHECK "UNIT 2",
001737 147400         AND 2,1           ;"SEEKING 2"
001740 122414         SUB# 1,0,SZR
001741 006253         EHLT
001742 006263         LOOPD
```

```
001743 006241 B12:    SETUP             ;CHECK UNIT SELECTION
001744 002056         LDA 0,C140K       ;START UNIT 3 SEEKING
001745 063033         DOC 0,DSKP        ;AND VERIFY THAT NO
001746 002070         LDA 0,C100K       ;OTHER UNIT SEEKS.
001747 061333         DOAP 1,DSKP       ;AC2=STATUS DURING SEEK
001750 073433         DIA 2,DSKP        ;AC1=BAD SEEKING STATUS
001751 024216         LDA 1,C360K       ;AC0=GOOD
001752 002066         LDA 0,KB8         ;CHECK "UNIT 3", AND
001753 147400         AND 2,1           ;"SEEKING 3"
001754 122414         SUBZ# 1,0,SZR
001755 006253         FHLT
001756 006263         LOOPD

001757 006241 B13:    SETUP             ;CHECK THE "CLEAR"
001760 152520         SUBZL 2,2         ;RESET OF "SEEKING 0".
001761 006230         SSEEK             ;START UNIT 0 SEEKING
001762 060233         NIOC DSKP         ;ISSUE (C) PULSE
001763 002071         LDA 0,KB5
001764 064433         DIA 1,DSKP        ;READ STATUS
001765 107404         AND 0,1,SZR       ;"SEEKING 0" BIT NOT
001766 006253         FHLT              ;CLEARED BY (C) PULSE
001767 006263         LOOPD

001770 006241 B14:    SETUP             ;CHECK THE "CLEAR"
001771 030060         LDA 2,KB14        ;RESET OF "SEEKING 1"
001772 006230         SSEEK             ;START UNIT 1 SEEKING.
001773 060233         NIOC DSKP         ;ISSUE (C) PULSE
001774 002073         LDA 0,KB6         ;READ STATUS
001775 064433         DIA 1,DSKP        ;"SEEKING 1" BIT NOT
001776 107404         AND 0,1,SZR       ;CLEARED BY (C) PULSE
001777 006253         FHLT
002000 006263         LOOPD

002001 006241 B15:    SETUP             ;CHECK THE "CLEAR" RESET
002002 030061         LDA 2,KB13        ;OF "SEEKING 2".
002003 006230         SSEEK             ;START UNIT 2 SEEKING.
002004 060233         NIOC DSKP         ;ISSUE (C) PULSE
002005 002067         LDA 0,KB7
002006 064433         DIA 1,DSKP        ;READ STATUS
002007 107404         AND 0,1,SZR       ;"SEEKING 2" BIT NOT
002010 006253         FHLT              ;RESET BY (C) PULSE
002011 006263         LOOPD
```

```
A 0025  .MAIN

02012 006241 B16:    SETUP            ;CHECK THE "CLEAR" RESET
02013 030062         LDA 2,KB12       ;OF "SEEKING 3"
02014 006230         SSEEK            ;START UNIT 3 SEEKING.
02015 006033         NIOC DSKP        ;ISSUE (C) PULSE
02016 020066         LDA 0,KB8
02017 064433         DIA 1,DSKP       ;READ STATUS
02020 107404         AND 0,1,SZR      ;"SEEKING 3" BIT NOT
02021 006253         EHLT             ;CLEARED BY (C) PULSE
02022 006263         LOOPD

02023 020151 B17:    LDA 0,NDSKS      ;ATTEMPT TO OBTAIN
02024 101203         MOVR 0,0,SNC     ;"SEEK DONE 0" FROM
02025 000407         JMP B18          ;A RECALIBRATE
02026 006237         JSR @ISET        ;(SKIP OVER IF NO UNIT 0)
02027 006231         RECL0            ;RECAL UNIT 0
02030 020054         LDA 0,KB1        ;AC1=STATUS
02031 123415         AND# 1,0,SNR     ;"ATTEN0" DID NOT SET
02032 006253         EHLT             ;"SEEK DONE 0"
02033 006243         LOOP

02034 020151 B18:    LDA 0,NDSKS      ;ATTEMPT TO OBTAIN
02035 101200         MOVR 0,0         ;"SEEK DONE 1" FROM
02036 101203         MOVR 0,0,SNC     ;A RECALIBRATE
02037 000407         JMP B19          ;(SKIP OVER IF NO UNIT 1)
02040 006237         JSR @ISET
02041 006232         RECL1            ;RECAL UNIT 1
02042 020074         LDA 0,KB2        ;AC1=STATUS
02043 123415         AND# 1,0,SNR     ;"ATTEN1" DID NOT SET
02044 006253         EHLT             ;"SEEK DONE 1"
02045 006243         LOOP

02046 020151 B19:    LDA 0,NDSKS      ;ATTEMPT TO OBTAIN
02047 024061         LDA 1,KB13       ;"SEEK DONE 2" FROM
02050 123405         AND 1,0,SNR      ;A RECALIBRATE
02051 000407         JMP B20          ;(SKIP OVER IF NO UNIT 2)
02052 006237         JSR @ISET
02053 006233         RECL2            ;RECAL UNIT 2
02054 020073         LDA 0,KB3        ;AC1=STATUS
02055 123415         AND# 1,0,SNR     ;"ATTEN2" DID NOT SET
02056 006253         EHLT             ;"SEEK DONE 2"
02057 006243         LOOP
```

```
02060 020151 B20:   LDA 0,NDSKS       ;ATTEMPT TO OBTAIN
02061 024062        LDA 1,KB12        ;"SEEK DONE 3" FROM
02062 123405        AND 1,0,SNR       ;A RECALIBRATE
02063 000407        JMP B21           ;(SKIP OVER IF NO UNIT 3)
02064 006237        JSR @ISET
02065 006234        RECL3             ;RECAL UNIT 3
02066 020072        LDA 0,KB4         ;AC1=STATUS
02067 123415        AND# 1,0,SNR      ;"ATTEN3" FAILED TO SET
02070 006253        FHLT              ;"SEEK DONE 3"
02071 006243        LOOP

02072 062677 B21:   IORST             ;DISK DRIVE WILL NOT TAKE
02073 030107        LDA 2,C5          ;SUCCESSIVE RECALIBRATES
02074 006227        WAIT              ;DO A DELAY

02075 020151        LDA 0,NDSKS       ;SEE IF "SEEK DONE 0"
02076 101203        MOVR 0,0,SNC      ;WILL RESET "SEEKING 0"
02077 000407        JMP B22           ;(SKIP IF NO UNIT 0)
02100 006237        JSR @ISET
02101 006231        RECL0             ;RECAL UNIT 0
02102 020071        LDA 0,KB5         ;AC1=STATUS
02103 123414        AND# 1,0,SZR      ;"SEEKING 0" SHOULD GET
02104 006253        FHLT              ;RESET BY "SEEK DONE 0"
02105 006243        LOOP

02106 020151 B22:   LDA 0,NDSKS       ;SEE IF "SEEK DONE 1"
02107 101200        MOVR 0,0          ;WILL RESET "SEEKING 1"
02110 101203        MOVR 0,0,SNC
02111 000407        JMP B23           ;(SKIP OVER IF NO UNIT 1)
02112 006237        JSR @ISET
02113 006232        RECL1             ;RECAL UNIT 1
02114 020070        LDA 0,KB6         ;AC1=STATUS
02115 123414        AND# 1,0,SZR      ;"SEEKING 1" SHOULD GET
02116 006253        FHLT              ;RESET BY "SEEK DONE 1"
02117 006243        LOOP
```

```
A 0027  .MAIN

 02120 020151 B23:    LDA 0,NDSKS      ;SEE IF "SEEK DONE 2"
 02121 024001         LDA 1,KB13       ;WILL RESET "SEEKING 2"
 02122 123405         AND 1,0,SNR
 02123 000407         JMP B24          ;(SKIP OVER IF NO UNIT 2)
 02124 006237         JSR @ISET
 02125 006233         RECL2            ;RECAL UNIT 2
 02126 020067         LDA 0,KB7        ;AC1=STATUS
 02127 123414         AND# 1,0,SZR     ;"SEEKING 2" SHOULD GET
 02130 006253         EHLT             ;RESET BY "SEEK DONE 2"
 02131 006243         LOOP

 02132 020151 B24:    LDA 0,NDSKS      ;SEE IF "SEEK DONE 3"
 02133 024062         LDA 1,KB12       ;WILL RESET "SEEKING 3"
 02134 123405         AND 1,0,SNR
 02135 000407         JMP B25          ;(SKIP IF NO UNIT 3)
 02136 006237         JSR @ISET
 02137 006234         RECL3            ;RECAL UNIT 3
 02140 020066         LDA 0,KB8        ;AC1=STATUS
 02141 123414         AND# 1,0,SZR     ;"SEEKING 3" SHOULD GET
 02142 006253         EHLT             ;RESET VIA "SEEK DONE 3"
 02143 006243         LOOP

 02144 062677 B25:    IORST            ;DISK DRIVE WILL NOT TAKE
 02145 030173         LDA 2,C15        ;SUCCESSIVE RECALIBRATES
 02146 006227         WAIT             ;WAIT 1.3 SEC

 02147 020151         LDA 0,NDSKS      ;ATTEMPT TO RESET "SEEK DONE 0"
 02150 101203         MOVR 0,0,SNC     ;WITH "DATOA"
 02151 000411         JMP B26          ;SKIP OVER IF NO UNIT 0
 02152 006237         JSR @ISET
 02153 006231         RECL0            ;SET "SEEK DONE 0" VIA
 02154 020054         LDA 0,KB1        ;RECALIBRATE
 02155 061033         DOA 0,DSKP       ;ATTEMPT RESET
 02156 064433         DIA 1,DSKP       ;READ STATUS
 02157 123414         AND# 1,0,SZR
 02160 006253         EHLT             ;"SEEK DONE 0" RESET FAILED
 02161 006243         LOOP

 02162 020151 B26:    LDA 0,NDSKS      ;ATTEMPT TO RESET "SEEK DONE 1"
 02163 101200         MOVR 0,0         ;WITH "DATOA"
 02164 101203         MOVR 0,0,SNC
 02165 000411         JMP B27          ;(SKIP OVER IF NO UNIT 1)
 02166 006237         JSR @ISET
 02167 006232         RECL1            ;SET "SEEK DONE 1"
 02170 020074         LDA 0,KB2        ;WITH A RECALIBRATE
 02171 061033         DOA 0,DSKP       ;ATTEMPT RESET
 02172 064433         DIA 1,DSKP       ;READ STATUS
 02173 123414         AND# 1,0,SZR
 02174 006253         EHLT             ;"SEEK DONE 1" RESET FAILED
 02175 006243         LOOP
```

```
02176 020151 B27:   LDA 0,NDSKS       ;ATTEMPT TO RESET "SEEK DONE 2"
02177 024061        LDA 1,KB13        ;WITH "DATOA"
02200 123405        AND 1,0,SNR
02201 000411        JMP B28           ;(SKIP OVER IF NO UNIT 2)
02202 006237        JSR @ISET
02203 006233        RECL2             ;SET "SEEK DONE 2"
02204 020073        LDA 0,KB3         ;WITH A RECALIBRATE
02205 061033        DOA 0,DSKP        ;ATTEMPT RESET
02206 064433        DIA 1,DSKP        ;READ STATUS
02207 123414        AND# 1,0,SZR
02210 006253        EHLT              ;"SEEK DONE 2" RESET FAILED
02211 006243        LOOP

02212 020151 B28:   LDA 0,NDSKS       ;ATTEMPT TO RESET "SEEK DONE 3"
02213 024062        LDA 1,KB12        ;WITH "DATOA"
02214 123405        AND 1,0,SNR
02215 000411        JMP B29           ;(SKIP OVER IF NO UNIT 3)
02216 006237        JSR @ISET
02217 006234        RECL3             ;SET "SEEK DONE 3"
02220 020072        LDA 0,KB4         ;WITH A RECALIBRATE
02221 061033        DOA 0,DSKP        ;ATTEMPT RESET
02222 064433        DIA 1,DSKP        ;READ STATUS
02223 123414        AND# 1,0,SZR
02224 006253        EHLT              ;"SEEK DONE 3" RESET FAILED
02225 006243        LOOP

02226 020151 B29:   LDA 0,NDSKS       ;ATTEMPT TO RESET "SEEK DONE 0"
02227 101203        MOVR 0,0,SNC      ;WITH A (C) PULSE
02230 000411        JMP B30           ;(SKIP OVER IF NO UNIT 0)
02231 006237        JSR @ISET
02232 006231        RECL0             ;SET "SEEK DONE 0" VIA RECAL
02233 020054        LDA 0,KB1
02234 060233        NIOC DSKP         ;ATTEMPT CLEAR
02235 064433        DIA 1,DSKP        ;READ STATUS
02236 123414        AND# 1,0,SZR
02237 006253        EHLT              ;"SEEK DONE 0" RESET FAILED
02240 006243        LOOP

02241 020151 B30:   LDA 0,NDSKS       ;ATTEMPT TO RESET "SEEK DONE 1"
02242 101200        MOVR 0,0          ;WITH A (C) PULSE
02243 101203        MOVR 0,0,SNC
02244 000411        JMP B31           ;(SKIP IF NO UNIT 1)
02245 006237        JSR @ISET
02246 006232        RECL1             ;SET "SEEK DONE 1" VIA RECAL
02247 020074        LDA 0,KB2
02250 060233        NIOC DSKP         ;ATTEMPT RESET
02251 064433        DIA 1,DSKP        ;READ STATUS
02252 123414        AND# 1,0,SZR
02253 006253        EHLT              ;"SEEK DONE 1" RESET FAILED
02254 006243        LOOP
```

```
02255 020151 B31:   LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 2"
02256 024061        LDA 1,KB13      ;WITH A (C) PULSE
02257 123405        AND 1,0,SNR
02260 000411        JMP B32         ;(SKIP OVER IF NO UNIT 2)
02261 006237        JSR @ISET
02262 006233        RECL2           ;SET "SEEK DONE 2" VIA RECAL
02263 020073        LDA 0,KB3
02264 060233        NIOC DSKP       ;ATTEMPT RESET
02265 064433        DIA 1,DSKP      ;READ STATUS
02266 123414        AND# 1,0,SZR
02267 006253        EHLT            ;"SEEK DONE 2" RESET FAILED
02270 006243        LOOP

02271 020151 B32:   LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 3"
02272 024062        LDA 1,KB12      ;WITH A (C) PULSE
02273 123405        AND 1,0,SNR
02274 000411        JMP B35         ;(SKIP OVER IF NO UNIT 3)
02275 006237        JSR @ISET
02276 006234        RECL3           ;SET "SEEK DONE 3" VIA RECAL
02277 020072        LDA 0,KB4
02300 060233        NIOC DSKP       ;ATTEMPT RESET
02301 064433        DIA 1,DSKP      ;READ STATUS
02302 123414        AND# 1,0,SZR
02303 006253        EHLT            ;"SEEK DONE 3" RESET FAILED
02304 006243        LOOP

02305 006241 B35:   SETUP           ;IDLE STATE STATUS CHECK
02306 064433        DIA 1,DSKP
02307 102620        SUBZR 0,0
02310 107414        AND# 0,1,SZR
02311 006242        EHALT           ;R/W DONE = 1
02312 006243        LOOP

02313 006241 B36:   SETUP           ;IDLE STATE STATUS CHECK
02314 064433        DIA 1,DSKP
02315 020175        LDA 0,C30
02316 123414        AND# 1,0,SZR
02317 006242        EHALT           ;ADDR ERR OR END CYL
02320 006243        LOOP
```

```
#2321 #06241 B37:   SETUP           ;IDLE STATE STATUS CHECK
#2322 #64433        DIA 1,DSKP
#2323 #20#61        LDA 0,KB13
#2324 107414        AND# #,1,SZR
#2325 #06242        EHALT           ;CHECK WORD ERROR = 1
#2326 #06243        LOOP

#2327 #06241 B38:   SETUP           ;IDLE STATE STATUS CHECK
#233# #64433        DIA 1,DSKP
#2331 #20117        LDA 0,C3
#2332 123414        AND# 1,0,SZR
#2333 #06242        EHALT           ;DATA LATE OR ERR
#2334 #06243        LOOP

#2335 #20151 B39:   LDA 0,NDSKS     ;SEE IF "SEEK DONE 0"
#2336 1#1203        MOVR #,#,SNC    ;WILL CAUSE INTERRUPT
#2337 #0#436        JMP B40         ;(SKIP IF NO UNIT 0)
#234# #06237        JSR @ISET
#2341 #06231        RFCL0           ;SET "SEEK DONE 0"
#2342 #06235        ITRWT           ;IS THERE AN INTERRUPT?
#2343 #06253        EHLT            ;NO, CHECK "INTERRUPT",
#2344 #06243        LOOP            ;"OP INT REQ".

#2345 #20151 B40:   LDA 0,NDSKS     ;SEE IF "SEEK DONE 1"
#2346 101240        MOVR #,#        ;WILL CAUSE INTERRUPT
#2347 101203        MOVR #,#,SNC
#235# #0#436        JMP B41         ;(SKIP IF NO UNIT 1)
#2351 #06237        JSR @ISET
#2352 #06232        RFCL1           ;SET "SEEK DONE 1"
#2353 #06235        ITRWT           ;IS THERE AN INTERRUPT ?
#2354 #06253        EHLT            ;NO, CHECK "INTERRUPT",
#2355 #06243        LOOP            ;"OP INT REQ".

#2356 #20151 B41:   LDA 0,NDSKS     ;SEE IF "SEEK DONE 2"
#2357 #24061        LDA 1,KB13      ;WILL CAUSE INTERRUPT
#236# 123405        AND 1,0,SNR
#2361 #0#436        JMP B42         ;(SKIP IF NO UNIT 2)
#2362 #06237        JSR @ISET
#2363 #06233        RFCL2           ;SET "SEEK DONE 2"
#2364 #06235        ITRWT           ;IS THERE AN INTERRUPT ?
#2365 #06253        EHLT            ;NO, CHECK "INTERRUPT",
#2366 #06243        LOOP            ;"OP INT REQ".

                    .EOT
```

```
00031  .MAIN

02367 02U151 B42:    LDA 0,NDSKS       ;SEE IF "SEEK DONE 3"
02370 024062         LDA 1,KB12        ;WILL CAUSE INTERRUPT
02371 123405         AND 1,0,SNR
02372 000406         JMP B43           ;(SKIP IF NO UNIT 3)
02373 006237         JSR @ISET
02374 006234         RECL3             ;SET "SEEK DONE 3"
02375 006235         ITRWT             ;IS THERE AN INTERRUPT ?
02376 006253         EHLT              ;NO, CHECK "INTERRUPT",
02377 006243         LOOP              ;"DP INT REQ".

02400 006240 B43:    SETP1             ;TEST SEEK TO CYLINDER 0
02401 020152         LDA 0,TESTU       ;(FIRST ATTEMPT AT COMPLETE SEEK)
02402 063033         DOC 0,DSKP        ;SELECT AN ACTIVE UNIT #
02403 020070         LDA 0,KB6
02404 061333         DOAP 0,DSKP       ;SEEK !
02405 030167         LDA 2,C5
02406 006227         WAIT              ;WAIT 500MS (OR UNTIL "DONE")
02407 020220         LDA 0,C74K        ;AC1=STATUS
02410 123415         AND# 1,0,SNR      ;NO SEEK DONE STATUS
02411 006242         EHALT             ;CHECK "SEEK","CONTROL 1","CYL",
02412 006243         LOOP              ;"HD+DIR","CONTROL 2", SEQUENCING.

02413 006240 B44:    SETP1             ;TEST SEEK TO CYLINDER 0
02414 020152         LDA 0,TESTU
02415 063033         DOC 0,DSKP        ;SELECT AN ACTIVE UNIT #
02416 020070         LDA 0,KB6
02417 061333         DOAP 0,DSKP       ;SEEK !
02420 030167         LDA 2,C5
02421 006227         WAIT              ;WAIT 500MS (OR UNTIL "DONE")
02422 020064         LDA 0,KB10        ;AC1=STATUS
02423 123414         AND# 1,0,SZR      ;SEEK ERROR STATUS
02424 006242         EHALT             ;CHECK "SEEK","CONTROL 1","CYL",
02425 006243         LOOP              ;"HD+DIR", "DIFF","CONTROL 2".

02426 006240 B45:    SETP1             ;TEST SEEK CYLINDER 77 (OCTAL)
02427 006236         GADSK
02430 006246         SEEK
02431 000077         77
02432 020065         LDA 0,KB9         ;AC1=ENDING STATUS
02433 030064         LDA 2,KB10
02434 107414         AND# 0,1,SZR
02435 147414         AND# 2,1,SZR      ;SEEK ERROR OR
02436 006242         EHALT             ;NO READY STATUS
02437 006243         LOOP

02440 006240 B46:    SETP1             ;TEST SEEK CYLINDER 300 (OCTAL)
02441 006236         GADSK
02442 006246         SEEK
02443 000300         300
02444 020065         LDA 0,KB9         ;AC1=ENDING STATUS
02445 030064         LDA 2,KB10
02446 107414         AND# 0,1,SZR
02447 147414         AND# 2,1,SZR      ;SEEK ERROR OR
02450 006242         EHALT             ;NO READY STATUS
02451 006243         LOOP
```

```
002452 006240 B47:     SETP1           ;CHOOSE AN ACTIVE DISK
002453 006236          GADSK           ;AND SET INTERRUPT VIA SEEK.
002454 006246          SEEK            ;CHECK FOR PROPER DISK
002455 000000          0               ;ADDRESS RESPONSE TO INTA
002456 006235          ITRWT           ;ALLOW INTERRUPT
002457 000401          JMP .+1
002460 024154          LDA 0,CDSK
002461 065477          INTA 1          ;GET THE ADDRESS
002462 122414          SUBZ# 1,0,SZR   ;INTA FAILED.
002463 006242          EHALT           ;AC0=GOOD
002464 006243          LOOP            ;AC1=BAD

002465 006240 B48:     SETP1           ;SEE IF (C) PULSE WILL
002466 006236          GADSK           ;CLEAR "DP INT REQ"
002467 006246          SEEK            ;SET "INTERRUPT" WITH
002470 000000          0               ;A SEEK
002471 060233          NIOC DSKP       ;CLEAR INT REQ
002472 006235          ITRWT           ;CHECK FOR INTERRUPT
002473 000402          JMP .+2         ;NO INTERRUPT, OK!!
002474 006242          EHALT           ;(C) THRU "CLEAR ALL" FAILS TO
002475 006243          LOOP            ;CLEAR "DP INT REQ".

002476 006240 B49:     SETP1           ;INSURE THAT "DP INT DISABLE"
002477 020067          LDA 1,KB7       ;INHIBITS INTERRUPTS
002500 062077          MSKO 0          ;SET DISABLE
002501 006236          GADSK           ;GET ACTIVE UNIT # IN (AC2)
002502 006246          SEEK            ;START A SEEK
002503 000000          0
002504 006235          ITRWT           ;IS THERE AN INTERRUPT ?
002505 000402          JMP .+2         ;NO, GOOD
002506 006242          EHALT           ;MSKO BIT 7 DID NOT SET "DP DISABLE"
002507 006243          LOOP            ;AND INHIBIT "DP INT REQ".

002510 006240 B50:     SETP1           ;INSURE THAT IORST WILL
002511 020067          LDA 0,KB7       ;CLEAR "DP INT DISABLE"
002512 062077          MSKO 0          ;SET "DP INT DISABLE"
002513 062677          IORST           ;CLEAR IT
002514 006236          GADSK           ;GET ACTIVE UNIT # IN (AC2)
002515 006246          SEEK            ;START A SEEK
002516 000000          0
002517 006235          ITRWT           ;IS THERE AN INTERRUPT
002520 006242          EHALT           ;NO, CHECK RESET OF "DP INT
002521 006243          LOOP            ;DISABLE".
```

```
A 0033  .MAIN

002522 002151 B51:     LDA 0,NDSKS      ;IF UNIT 0 NOT AVAILABLE
002523 101203          MOVR 0,0,SNC     ;GO ON TO UNIT 1
002524 000423          JMP B53          ;(UNIT 0 NOT BEING TESTED)
002525 006241          SETUP            ;IDLE STATE STATUS CHECK
002526 152520          SUBZL 2,2        ;SELECT THE ADAPTER
002527 006246          SEEK             ;WITH A SEEK 0
002530 000000          0
002531 064433          DIA 1,DSKP       ;READ STATUS
002532 002065          LDA 0,KB9
002533 123415          AND# 1,0,SNR
002534 006242          EHALT            ;NO READY STATUS, UNIT 0
002535 006243          LOOP

002536 006241 B52:     SETUP            ;IDLE STATE STATUS CHECK
002537 152520          SUBZL 2,2        ;SELECT THE ADAPTER
002540 006246          SEEK             ;WITH A SEEK 0
002541 000000          0
002542 064433          DIA 1,DSKP       ;READ STATUS
002543 002064          LDA 0,KB10
002544 123414          AND# 1,0,SZR
002545 006242          EHALT            ;SEEK ERROR STATUS, UNIT 0
002546 006243          LOOP


002547 002151 B53:     LDA 0,NDSKS      ;IF UNIT 1 IS NOT
002550 101200          MOVR 0,0         ;AVAILABLE, GO ON TO UNIT 2
002551 101203          MOVR 0,0,SNC
002552 000423          JMP B55          ;SKIP THIS TEST
002553 006241          SETUP            ;IDLE STATE STATUS CHECK
002554 030060          LDA 2,KB14       ;SELECT THE ADAPTER
002555 006246          SEEK             ;WITH A SEEK 0
002556 000000          0
002557 064433          DIA 1,DSKP       ;READ STATUS
002560 002065          LDA 0,KB9
002561 123415          AND# 1,0,SNR
002562 006242          EHALT            ;NO READY STATUS, UNIT 1
002563 006243          LOOP

002564 006241 B54:     SETUP            ;IDLE STATE STATUS CHECK
002565 030060          LDA 2,KB14
002566 006246          SEEK             ;SELECT THE ADAPTER
002567 000000          0                ;WITH A SEEK 0
002570 064433          DIA 1,DSKP       ;READ STATUS
002571 002064          LDA 0,KB10
002572 123414          AND# 1,0,SZR
002573 006242          EHALT            ;SEEK ERROR STATUS, UNIT 1
002574 006243          LOOP
```

```
02575 002151 B55:    LDA 0,NDSKS      ;IF UNIT 2 IS NOT AVAILABLE
02576 024061         LDA 1,KB13       ;GO ON TO UNIT 3
02577 127405         AND 0,1,SNR
02600 000423         JMP B57          ;SKIP THIS TEST, NO UNIT 2
02601 006241         SETUP            ;IDLE STATE STATUS CHECK
02602 030061         LDA 2,KB13       ;SELECT THE ADAPTER
02603 006246         SEEK             ;WITH A SEEK 0
02604 000000         0
02605 064433         DIA 1,DSKP       ;READ STATUS
02606 002005         LDA 0,KB9
02607 123415         AND# 1,0,SNR
02610 006242         FHALT            ;NO READY STATUS, UNIT 2
02611 006243         LOOP

02612 006241 B56:    SETUP            ;IDLE STATE STATUS CHECK
02613 030061         LDA 2,KB13       ;SELECT THE ADAPTER
02614 006246         SEEK             ;WITH A SEEK 0
02615 000000         0
02616 064433         DIA 1,DSKP       ;READ STATUS
02617 002004         LDA 0,KB10
02620 123414         AND# 1,0,SZR
02621 006242         FHALT            ;SEEK ERROR STATUS, UNIT 2
02622 006243         LOOP

02623 002151 B57:    LDA 0,NDSKS      ;IF UNIT 3 IS NOT AVAILABLE
02624 024062         LDA 1,KB12       ;GO TO NEXT TEST.
02625 123405         AND 1,0,SNR      ;
02626 000423         JMP D1           ;SKIP, NO UNIT 3
02627 006241         SETUP            ;IDLE STATE STATUS CHECK
02630 030062         LDA 2,KB12       ;SELECT THE ADAPTER
02631 006246         SEEK             ;WITH A SEEK 0
02632 000000         0
02633 064433         DIA 1,DSKP       ;READ STATUS
02634 002005         LDA 0,KB9
02635 123415         AND# 1,0,SNR
02636 006242         FHALT            ;NO READY STATUS, UNIT 3
02637 006243         LOOP

02640 006241 B58:    SETUP            ;IDLE STATE STATUS CHECK
02641 030062         LDA 2,KB12       ;SELECT THE ADAPTER
02642 006246         SEEK             ;WITH A SEEK 0
02643 000000         0
02644 064433         DIA 1,DSKP       ;READ STATUS
02645 002004         LDA 0,KB10
02646 123414         AND# 1,0,SZR
02647 006242         FHALT            ;SEEK ERROR STATUS, UNIT 3
02650 006243         LOOP
```

```
02651 006241 D1:    SETUP           ;CHECK THE FIRST 2 DATA
02652 020152        LDA 0,TESTU     ;CHANNEL CYCLES OF A WRITE
02653 063033        DOC 0,DSKP      ;(FIRST ATTEMPT AT WRITE)
02654 030160        LDA 2,BUFF
02655 072033        DOB 2,DSKP      ;LOAD CA REG.
02656 020067        LDA 0,KB7
02657 061133        DOAS 0,DSKP     ;WRITE !!
02660 006165        JSR @STALL      ;ALLOW TIME FOR DCH CYCLES
02661 060233        NIOC DSKP       ;CLEAR THE WRITE OPERATION
02662 061433        DIB 0,DSKP      ;AC0=ENDING MEMORY ADDRESS
02663 024060        LDA 1,KB14      ;AC2=STARTING MEMORY ADDRESS
02664 147000        ADD 2,1         ;2 DCH CYCLES SHOULD HAVE
02665 122414        SUB# 1,0,SZR    ;OCCURRED
02666 006242        EHALT           ;AC0=BAD, AC1=GOOD
02667 006243        LOOP            ;CHECK THE SEQUENCE "DP FLAG"-
                                    ;"REQ1"-"2ND REQ". "DP FLAG"
                                    ;SETS "DP DCH REQ" WHICH STAYS
                                    ;ON FOR 2 MEMORY CYCLES.

02670 006241 D2:    SETUP           ;CHECK THE FIRST 2 DATA
02671 020152        LDA 0,TESTU     ;CHANNEL CYCLES OF A WRITE
02672 063033        DOC 0,DSKP
02673 030160        LDA 2,BUFF
02674 072033        DOB 2,DSKP      ;SET MEM ADDR REG.
02675 024140        LDA 1,C2525
02676 045000        STA 1,0,2       ;DATA = 052525
02677 045001        STA 1,1,2       ; 2 WORDS
02700 020067        LDA 0,KB7
02701 061133        DOAS 0,DSKP     ;WRITE !!
02702 006165        JSR @STALL      ;ALLOW TIME FOR 2 DCH'S
02703 060233        NIOC DSKP       ;CLEAR THE WRITE
02704 021000        LDA 0,0,2       ;IS THE DATA STILL IN MEMORY
02705 031001        LDA 2,1,2       ;IF NOT A DCHI MAY HAVE
02706 106415        SUB# 0,1,SNR    ;BEEN PERFORMED INSTEAD
02707 146414        SUB# 2,1,SZR    ;OF A DCHO
02710 006242        EHALT           ;AC0&AC2=BAD WORDS
02711 006243        LOOP            ;AC1=GOOD
```

```
A 0036  .MAIN

002712 006241 D3:     SETUP             ;CHECK PROPER CA REGISTER
002713 020152         LDA 0,TESTU       ;INCREMENT BY ALLOWING
002714 063033         DOC 0,DSKP        ;THE FIRST 2 DATA CHANNEL
002715 102400         SUB 0,0           ;CYCLES AT THE BEGINNING
002716 062033         DOB 0,DSKP        ;OF A WRITE.
002717 024067         LDA 1,KB7
002720 065133         DOAS 1,DSKP       ; WRITE !!
002721 006165         JSR @STALL
002722 063233         NIOC DSKP         ;STOP THE WRITE
002723 030060         LDA 2,KB14        ;AC0=STARTING MEMORY ADDRESS
002724 065433         DIB 1,DSKP        ;AC1=ACTUAL ENDING MEMORY
002725 113000         ADD 0,2           ;     ADDRESS
002726 146414         SUB# 2,1,SZR      ;AC2=CORRECT ENDING MEMORY
002727 006242         EHALT             ;     ADDRESS
002730 006243         LOOP

002731 006241 D4:     SETUP             ;CHECK PROPER CA REGISTER
002732 020152         LDA 0,TESTU       ;INCREMENT BY ALLOWING
002733 063033         DOC 0,DSKP        ;THE FIRST 2 DATA CHANNEL
002734 102520         SUBZL 0,0         ;CYCLES AT THE BEGINNING
002735 062033         DOB 0,DSKP        ;OF A WRITE.
002736 024067         LDA 1,KB7
002737 065133         DOAS 1,DSKP       ; WRITE !!
002740 006165         JSR @STALL
002741 063233         NIOC DSKP         ;STOP THE WRITE
002742 030060         LDA 2,KB14        ;AC0=STARTING MEMORY ADDRESS
002743 065433         DIB 1,DSKP        ;AC1=ACTUAL ENDING MEMORY
002744 113000         ADD 0,2           ;     ADDRESS
002745 146414         SUB# 2,1,SZR      ;AC2=CORRECT ENDING MEMORY
002746 006242         EHALT             ;     ADDRESS
002747 006243         LOOP

002750 006241 D5:     SETUP             ;CHECK PROPER CA REGISTER
002751 020152         LDA 0,TESTU       ;INCREMENT BY ALLOWING
002752 063033         DOC 0,DSKP        ;THE FIRST 2 DATA CHANNEL
002753 024117         LDA 0,C3          ;CYCLES AT THE BEGINNING
002754 062033         DOB 0,DSKP        ;OF A WRITE.
002755 024067         LDA 1,KB7
002756 065133         DOAS 1,DSKP       ; WRITE !!
002757 006165         JSR @STALL
002760 063233         NIOC DSKP         ;STOP THE WRITE
002761 030060         LDA 2,KB14        ;AC0=STARTING MEMORY ADDRESS
002762 065433         DIB 1,DSKP        ;AC1=ACTUAL ENDING MEMORY
002763 113000         ADD 0,2           ;     ADDRESS
002764 146414         SUB# 2,1,SZR      ;AC2=CORRECT ENDING MEMORY
002765 006242         EHALT             ;     ADDRESS
002766 006243         LOOP
```

```
02767 000241 D6:    SETUP           ;CHECK PROPER CA REGISTER
02770 020152        LDA 0,TESTIO    ;INCREMENT BY ALLOWING
02771 063033        DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
02772 020120        LDA 0,C7        ;CYCLES AT THE BEGINNING
02773 062033        DOB 0,DSKP      ;OF A WRITE.
02774 024067        LDA 1,KB7
02775 065133        DOAS 1,DSKP     ; WRITE !!
02776 000165        JSR @STALL
02777 060233        NIOC DSKP       ;STOP THE WRITE
03000 030060        LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
03001 065433        DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
03002 113000        ADD 0,2         ;     ADDRESS
03003 146414        SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
03004 000242        EHALT           ;     ADDRESS
03005 000243        LOOP

03006 000241 D7:    SETUP           ;CHECK PROPER CA REGISTER
03007 020152        LDA 0,TESTIO    ;INCREMENT BY ALLOWING
03010 063033        DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
03011 020121        LDA 0,C17       ;CYCLES AT THE BEGINNING
03012 062033        DOB 0,DSKP      ;OF A WRITE.
03013 024067        LDA 1,KB7
03014 065133        DOAS 1,DSKP     ; WRITE !!
03015 000165        JSR @STALL
03016 060233        NIOC DSKP       ;STOP THE WRITE
03017 030060        LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
03020 065433        DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
03021 113000        ADD 0,2         ;     ADDRESS
03022 146414        SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
03023 000242        EHALT           ;     ADDRESS
03024 000243        LOOP

03025 000241 D8:    SETUP           ;CHECK PROPER CA REGISTER
03026 020152        LDA 0,TESTIO    ;INCREMENT BY ALLOWING
03027 063033        DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
03030 020122        LDA 0,C37       ;CYCLES AT THE BEGINNING
03031 062033        DOB 0,DSKP      ;OF A WRITE.
03032 024067        LDA 1,KB7
03033 065133        DOAS 1,DSKP     ; WRITE !!
03034 000165        JSR @STALL
03035 060233        NIOC DSKP       ;STOP THE WRITE
03036 030060        LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
03037 065433        DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
03040 113000        ADD 0,2         ;     ADDRESS
03041 146414        SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
03042 000242        EHALT           ;     ADDRESS
03043 000243        LOOP
```

```
003244 006241 D9:    SETUP              ;CHECK PROPER CA REGISTER
003245 020152        LDA 0,TESTU        ;INCREMENT BY ALLOWING
003246 063033        DOC 0,DSKP         ;THE FIRST 2 DATA CHANNEL
003247 020123        LDA 0,C77          ;CYCLES AT THE BEGINNING
003250 062033        DOB 0,DSKP         ;OF A WRITE.
003251 024067        LDA 1,KB7
003252 065133        DOAS 1,DSKP        ; WRITE !!
003253 006165        JSR @STALL
003254 061233        NIOC DSKP          ;STOP THE WRITE
003255 030060        LDA 2,KB14         ;AC0=STARTING MEMORY ADDRESS
003256 065433        DIB 1,DSKP         ;AC1=ACTUAL ENDING MEMORY
003257 113000        ADD 0,2            ;     ADDRESS
003260 146414        SUBZ 2,1,SZR       ;AC2=CORRECT ENDING MEMORY
003261 006242        EHALT              ;     ADDRESS
003262 006243        LOOP

003263 006241 D10:   SETUP              ;CHECK PROPER CA REGISTER
003264 020152        LDA 0,TESTU        ;INCREMENT BY ALLOWING
003265 063033        DOC 0,DSKP         ;THE FIRST 2 DATA CHANNEL
003266 020124        LDA 0,C177         ;CYCLES AT THE BEGINNING
003267 062033        DOB 0,DSKP         ;OF A WRITE.
003270 024067        LDA 1,KB7
003271 065133        DOAS 1,DSKP        ; WRITE !!
003272 006165        JSR @STALL
003273 061233        NIOC DSKP          ;STOP THE WRITE
003274 030060        LDA 2,KB14         ;AC0=STARTING MEMORY ADDRESS
003275 065433        DIB 1,DSKP         ;AC1=ACTUAL ENDING MEMORY
003276 113000        ADD 0,2            ;     ADDRESS
003277 146414        SUBZ 2,1,SZR       ;AC2=CORRECT ENDING MEMORY
003300 006242        EHALT              ;     ADDRESS
003301 006243        LOOP

003302 006241 D11:   SETUP              ;CHECK PROPER CA REGISTER
003303 020152        LDA 0,TESTU        ;INCREMENT BY ALLOWING
003304 063033        DOC 0,DSKP         ;THE FIRST 2 DATA CHANNEL
003305 020125        LDA 0,C377         ;CYCLES AT THE BEGINNING
003306 062033        DOB 0,DSKP         ;OF A WRITE.
003307 024067        LDA 1,KB7
003310 065133        DOAS 1,DSKP        ; WRITE !!
003311 006165        JSR @STALL
003312 061233        NIOC DSKP          ;STOP THE WRITE
003313 030060        LDA 2,KB14         ;AC0=STARTING MEMORY ADDRESS
003314 065433        DIB 1,DSKP         ;AC1=ACTUAL ENDING MEMORY
003315 113000        ADD 0,2            ;     ADDRESS
003316 146414        SUBZ 2,1,SZR       ;AC2=CORRECT ENDING MEMORY
003317 006242        EHALT              ;     ADDRESS
003320 006243        LOOP
```

```
03121 000241 D12:   SETUP           ;CHECK PROPER CA REGISTER
03122 020152        LDA 0,TESTU     ;INCREMENT BY ALLOWING
03123 063033        DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
03124 020126        LDA 0,C777      ;CYCLES AT THE BEGINNING
03125 062033        DOB 0,DSKP      ;OF A WRITE.
03126 024067        LDA 1,KB7
03127 065133        DOAS 1,DSKP     ; WRITE !!
03130 000165        JSR @STALL
03131 060233        NIOC DSKP       ;STOP THE WRITE
03132 030060        LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
03133 065433        DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
03134 113000        ADD 0,2         ;    ADDRESS
03135 146414        SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
03136 000242        EHALT           ;    ADDRESS
03137 000243        LOOP

03140 000241 D13:   SETUP           ;CHECK PROPER CA REGISTER
03141 020152        LDA 0,TESTU     ;INCREMENT BY ALLOWING
03142 063033        DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
03143 020127        LDA 0,C1777     ;CYCLES AT THE BEGINNING
03144 062033        DOB 0,DSKP      ;OF A WRITE.
03145 024067        LDA 1,KB7
03146 065133        DOAS 1,DSKP     ; WRITE !!
03147 000165        JSR @STALL
03150 060233        NIOC DSKP       ;STOP THE WRITE
03151 030060        LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
03152 065433        DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
03153 113000        ADD 0,2         ;    ADDRESS
03154 146414        SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
03155 000242        EHALT           ;    ADDRESS
03156 000243        LOOP

03157 000241 D14:   SETUP           ;CHECK PROPER CA REGISTER
03160 020152        LDA 0,TESTU     ;INCREMENT BY ALLOWING
03161 063033        DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
03162 020130        LDA 0,C3777     ;CYCLES AT THE BEGINNING
03163 062033        DOB 0,DSKP      ;OF A WRITE.
03164 024067        LDA 1,KB7
03165 065133        DOAS 1,DSKP     ; WRITE !!
03166 000165        JSR @STALL
03167 060233        NIOC DSKP       ;STOP THE WRITE
03170 030060        LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
03171 065433        DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
03172 113000        ADD 0,2         ;    ADDRESS
03173 146414        SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
03174 000242        EHALT           ;    ADDRESS
03175 000243        LOOP
```

```
003176 000241 D15:    SETUP             ;CHECK PROPER CA REGISTER
003177 020152         LDA 0,TESTU       ;INCREMENT BY ALLOWING
003200 063033         DOC 0,DSKP        ;THE FIRST 2 DATA CHANNEL
003201 020131         LDA 0,C7777       ;CYCLES AT THE BEGINNING
003202 062033         DOB 0,DSKP        ;OF A WRITE.
003203 024067         LDA 1,KB7
003204 065133         DOAS 1,DSKP       ; WRITE !!
003205 000165         JSR @STALL
003206 060233         NIOC DSKP         ;STOP THE WRITE
003207 030060         LDA 2,KB14        ;AC0=STARTING MEMORY ADDRESS
003210 065433         DIB 1,DSKP        ;AC1=ACTUAL ENDING MEMORY
003211 113033         ADD 0,2           ;     ADDRESS
003212 146414         SUB# 2,1,SZR      ;AC2=CORRECT ENDING MEMORY
003213 000242         EHALT             ;     ADDRESS
003214 000243         LOOP

003215 000241 D16:    SETUP             ;CHECK PROPER CA REGISTER
003216 020152         LDA 0,TESTU       ;INCREMENT BY ALLOWING
003217 063033         DOC 0,DSKP        ;THE FIRST 2 DATA CHANNEL
003220 020132         LDA 0,C017        ;CYCLES AT THE BEGINNING
003221 062033         DOB 0,DSKP        ;OF A WRITE.
003222 024067         LDA 1,KB7
003223 065133         DOAS 1,DSKP       ; WRITE !!
003224 000165         JSR @STALL
003225 060233         NIOC DSKP         ;STOP THE WRITE
003226 030060         LDA 2,KB14        ;AC0=STARTING MEMORY ADDRESS
003227 065433         DIB 1,DSKP        ;AC1=ACTUAL ENDING MEMORY
003230 113000         ADD 0,2           ;     ADDRESS
003231 146414         SUB# 2,1,SZR      ;AC2=CORRECT ENDING MEMORY
003232 000242         EHALT             ;     ADDRESS
003233 000243         LOOP

003234 000241 D17:    SETUP             ;CHECK PROPER CA REGISTER
003235 020152         LDA 0,TESTU       ;INCREMENT BY ALLOWING
003236 063033         DOC 0,DSKP        ;THE FIRST 2 DATA CHANNEL
003237 020133         LDA 0,C037        ;CYCLES AT THE BEGINNING
003240 062033         DOB 0,DSKP        ;OF A WRITE.
003241 024067         LDA 1,KB7
003242 065133         DOAS 1,DSKP       ; WRITE !!
003243 000165         JSR @STALL
003244 060233         NIOC DSKP         ;STOP THE WRITE
003245 030060         LDA 2,KB14        ;AC0=STARTING MEMORY ADDRESS
003246 065433         DIB 1,DSKP        ;AC1=ACTUAL ENDING MEMORY
003247 113000         ADD 0,2           ;     ADDRESS
003250 146414         SUB# 2,1,SZR      ;AC2=CORRECT ENDING MEMORY
003251 000242         EHALT             ;     ADDRESS
003252 000243         LOOP
```

```
03253 006241 D18:    SETUP              ;CHECK PROPER CA REGISTER
03254 020152         LDA 0,TESTII       ;INCREMENT BY ALLOWING
03255 063033         DOC 0,DSKP         ;THE FIRST 2 DATA CHANNEL
03256 102220         ADCZR 0,0          ;CYCLES AT THE BEGINNING
03257 062033         DOA 0,DSKP         ;OF A WRITE.
03260 024067         LDA 1,KB7
03261 065133         DOAS 1,DSKP        ; WRITE !!
03262 006165         JSR @STALL
03263 060233         NIOC DSKP          ;STOP THE WRITE
                                        ;AC0=STARTING MEMORY ADDRESS
03264 065433         DIB 1,DSKP         ;AC1=ACTUAL ENDING MEMORY
03265 152520         SUBZL 2,2          ;     ADDRESS
03266 146414         SUB# 2,1,SZR       ;AC2=CORRECT ENDING MEMORY
03267 006242         EHALT              ;     ADDRESS
03270 006243         LOOP

03271 006241 D19:    SETUP              ;CHECK PROPER CA REGISTER
03272 020152         LDA 0,TESTII       ;INCREMENT BY ALLOWING
03273 063033         DOC 0,DSKP         ;THE FIRST 2 DATA CHANNEL
03274 102000         ADC 0,0            ;CYCLES AT THE BEGINNING
03275 062033         DOA 0,DSKP         ;OF A WRITE.
03276 024067         LDA 1,KB7
03277 065133         DOAS 1,DSKP        ; WRITE !!
03300 006165         JSR @STALL
03301 060233         NIOC DSKP          ;STOP THE WRITE
03302 065433         DIB 1,DSKP         ;AC0=STARTING MEMORY ADDRESS
03303 152620         SUBZR 2,2          ;AC1=ACTUAL ENDING MEMORY
03304 151400         INC 2,2            ;     ADDRESS
03305 146414         SUB# 2,1,SZR       ;AC2=CORRECT ENDING MEMORY
03306 006242         EHALT              ;     ADDRESS
03307 006243         LOOP


                     .EOT
```

```
Ø331Ø ØØ624Ø D2Ø:    SETP1           ;ATTEMPT A COMPLETE WRITE
Ø3311 Ø2416Ø         LDA 1,BUFF      ;(FIRST WRITE WITH WC OVFLO)
Ø3312 Ø66Ø33         DOB 1,DSKP      ;ONE SECTOR
Ø3313 Ø2Ø152         LDA Ø,TESTU
Ø3314 Ø24121         LDA 1,C17
Ø3315 123Ø2Ø         ADD 1,Ø         ;UNIT # + SECT CNT
Ø3316 Ø63Ø33         DOC Ø,DSKP      ;SELECT UNIT
Ø3317 Ø2ØØ67         LDA Ø,KB7
Ø332Ø Ø61133         DOAS Ø,DSKP     ;WRITE !!
Ø3321 ØØ6235         ITRPT           ;WAIT 1ØØMS OR UNTIL INTERRUPT
Ø3322 ØØ6242         EHALT           ;TIMEOUT, NO INTERRUPT
Ø3323 ØØ6243         LOOP

Ø3324 ØØ624Ø D21:    SETP1           ;ATTEMPT TO CLEAR "DP DONE"
Ø3325 Ø2416Ø         LDA 1,BUFF      ;WITH (C) PULSE.
Ø3326 Ø66Ø33         DOB 1,DSKP
Ø3327 Ø2Ø152         LDA Ø,TESTU
Ø333Ø Ø24121         LDA 1,C17       ;SET "DP DONE" WITH A ONE
Ø3331 123ØØØ         ADD 1,Ø         ;SECTOR WRITE.
Ø3332 Ø63Ø33         DOC Ø,DSKP      ;SELECT UNIT
Ø3333 Ø2ØØ67         LDA Ø,KB7
Ø3334 Ø61133         DOAS Ø,DSKP     ;WRITE !!
Ø3335 15252Ø         SUBZL 2,2
Ø3336 ØØ6227         WAIT            ;WAIT 1ØØMS OR UNTIL DONE
Ø3337 Ø6Ø233         NIOC DSKP       ;ATTEMPT CLEAR
Ø334Ø Ø63733         SKPDZ DSKP      ;(C) PULSE DOES NOT
Ø3341 ØØ6242         EHALT           ;CLEAR "DP DONE" FF.
Ø3342 ØØ6243         LOOP

Ø3343 ØØ624Ø D22:    SETP1           ;ATTEMPT TO RESET
Ø3344 Ø2416Ø         LDA 1,BUFF      ;"DP DONE" WITH "DP DATOA"
Ø3345 Ø66Ø33         DOB 1,DSKP      ;AND "DATA Ø".
Ø3346 Ø2Ø152         LDA Ø,TESTU
Ø3347 Ø24121         LDA 1,C17       ;SET "DP DONE" WITH A 1 SECT.
Ø335Ø 123ØØØ         ADD 1,Ø         ;WRITE.
Ø3351 Ø63Ø33         DOC Ø,DSKP      ;SELECT UNIT
Ø3352 Ø2ØØ67         LDA Ø,KB7
Ø3353 Ø61133         DOAS Ø,DSKP     ;WRITE !!
Ø3354 15252Ø         SUBZL 2,2
Ø3355 ØØ6227         WAIT            ;WAIT 1ØØMS OR UNTIL DONE
Ø3356 1Ø262Ø         SUBZR Ø,Ø
Ø3357 Ø61Ø33         DOA Ø,DSKP      ;ATTEMPT RESET
Ø336Ø Ø63733         SKPDZ DSKP      ;"DP DATOA" AND "DATA Ø" DOES
Ø3361 ØØ6242         EHALT           ;NOT CLEAR "DP DONE".
Ø3362 ØØ6243         LOOP
```

```
003363 006240 D23:    SETP1             ;CHECK BUSY-DONE
003364 024160         LDA 1,BUFF        ;FOLLOWING A 1 SECTOR WRITE
003365 060033         DOB 1,DSKP
003366 020152         LDA 0,TESTU
003367 024121         LDA 1,C17
003370 123000         ADD 1,0
003371 063033         DOC 0,DSKP        ;SELECT UNIT / 1 SECT XFER
003372 020067         LDA 0,KB7
003373 061133         DOAS 0,DSKP       ;WRITE !!
003374 152520         SUBZL 2,2
003375 006227         WAIT              ;WAIT 100MS (OR UNTIL "DONE")
003376 063733         SKPDZ DSKP
003377 063533         SKPBZ DSKP        ;ERROR IF "DP DONE" = 0
003400 006242         EHALT             ;OR "DP BUSY" = 1
003401 006243         LOOP

003402 006240 D24:    SETP1             ;SEE IF A 1 SECTOR
003403 024160         LDA 1,BUFF        ;WRITE CAUSES "DATA LATE"
003404 060033         DOB 1,DSKP
003405 020152         LDA 0,TESTU
003406 024121         LDA 1,C17
003407 123000         ADD 1,0
003410 063033         DOC 0,DSKP        ;SELECT UNIT / 1 SECT XFER
003411 020067         LDA 0,KB7
003412 061133         DOAS 0,DSKP       ;WRITE !!
003413 152520         SUBZL 2,2
003414 006227         WAIT              ;WAIT 100MS, (OR UNTIL "DONE")
003415 064433         DIA 1,DSKP
003416 020000         LDA 0,KB14
003417 107414         ANDZ# 0,1,SZR     ;"REQ1" DOES NOT CLEAR
003420 006242         EHALT             ;"DP FLAG".  RESULTS IN
003421 006243         LOOP              ;"DATA LATE" STATUS.

003422 006240 D25:    SETP1             ;CHECK ENDING MEMORY
003423 024160         LDA 1,BUFF        ;ADDRESS (CA REGISTER)
003424 060033         DOB 1,DSKP        ;FOLLOWING A 1 SECTOR WRITE
003425 020152         LDA 0,TESTU
003426 024121         LDA 1,C17
003427 123000         ADD 1,0
003430 063033         DOC 0,DSKP        ;SELECT UNIT / 1 SECT XFER
003431 020067         LDA 0,KB7
003432 061133         DOAS 0,DSKP       ;WRITE !!
003433 152520         SUBZL 2,2
003434 006227         WAIT              ;WAIT 100MS ( OR UNTIL "DONE" )
003435 065433         DIB 1,DSKP        ;AC1=ENDING MEMORY ADDRESS
003436 020160         LDA 0,BUFF        ;CORRECT ENDING ADDRESS =
003437 030211         LDA 2,C402        ;(BUFF)+400+2, IN (AC0)
003440 143000         ADD 2,0
003441 122414         SUBZ# 1,0,SZR     ;ENDING MEM ADDR ERROR.
003442 006242         EHALT             ;AC0=GOOD
003443 006243         LOOP              ;AC1=BAD
```

```
03444 006240 D26:    SETP1              ;CHECK FOR PROPER ENDING
03445 024160         LDA 1,BUFF         ;DISK ADDRESS FOLLOWING
03446 060033         DOB 1,DSKP         ;1 SECTOR WRITE ON HEAD 0
03447 020152         LDA 0,TESTII       ;SECTOR 0
03450 024121         LDA 1,C17          ;"INC SC" CAUSES "SC1-SC8"
03451 123000         ADD 1,0            ;TO GO FROM 1111 TO 0000
03452 063033         DOC 0,DSKP         ;AND SETS "S1".
03453 020067         LDA 0,KB7
03454 061133         DOAS 0,DSKP        ;WRITE !!
03455 152520         SUBZL 2,2
03456 006227         WAIT               ;WAIT 100MS (OR UNTIL "DONE")
03457 060433         DIC 1, DSKP        ;READ ENDING DISK ADDRESS
03460 030135         LDA 2,CM37
03461 147400         AND 2,1            ;MASK OUT UNIT #
03462 020063         LDA 0,KB11         ;SECTOR = 1
03463 122414         SUB# 1,0,SZR
03464 000242         EHALT              ;ENDING DISK ADDRESS NOT
03465 000243         LOOP               ;HEAD=0, SECTOR=1, SC=0

03466 006240 D27:    SETP1              ;ATTEMPT A 2 SECTOR
03467 020160         LDA 0,BUFF         ;WRITE.
03470 062033         DOB 0,DSKP         ;FIRST ATTEMPT AT A WRITE
03471 020152         LDA 0,TESTII       ;LONGER THAN 1 SECTOR
03472 024174         LDA 1,C16
03473 123000         ADD 1,0
03474 063033         DOC 0,DSKP         ;SELECT UNIT / 2 SECTORS
03475 020067         LDA 0,KB7
03476 061133         DOAS 0,DSKP        ;WRITE !!
03477 152520         SUBZL 2,2
03500 006227         WAIT               ;WAIT 100MS (OR UNTIL "DONE")
03501 060433         DIC 1,DSKP         ;READ ENDING DISK ADDRESS
03502 020133         LDA 0,CM37         ;MASK OFF UNIT #
03503 147400         AND 0,1
03504 020064         LDA 0,KB10
03505 122414         SUB# 1,0,SZR       ;ENDING DISK ADDRESS NOT
03506 000242         EHALT              ;HEAD=0, SECT=2, SC=0
03507 000243         LOOP               ;AC0=GOOD ADDR, AC1=BAD

03510 006240 D28:    SETP1              ;ATTEMPT A 2 SECTOR WRITE
03511 020160         LDA 0,BUFF
03512 062033         DOB 0,DSKP
03513 020152         LDA 0,TESTII
03514 024174         LDA 1,C16
03515 123000         ADD 1,0
03516 063033         DOC 0,DSKP         ;SELECT UNIT / 2 SECTORS
03517 020067         LDA 0,KB7
03520 061133         DOAS 0,DSKP        ;WRITE !!
03521 152520         SUBZL 2,2
03522 006227         WAIT               ;WAIT 100MS )OR UNTIL "DONE")
03523 064433         DIA 1,DSKP         ;READ STATUS
03524 063633         SKPDN DSKP         ;NO "DP DONE"
03525 000242         EHALT              ;FOLLOWING 2 SECTOR WRITE
03526 000243         LOOP               ;(AC1)=STATUS REG.
```

```
003527 006240 D29:    SETP1           ;INSURE THAT "INC HEAD"
003530 002160         LDA 0,BUFF      ;FUNCTIONS PROPERLY
003531 062033         DOB 0,DSKP      ;TRANSFER 7 SECTORS OR 13
003532 002152         LDA 0,TESTII    ;SECTORS DEPENDING UPON
003533 003150         LDA 2,DTYPE     ;THE DISK TYPE
003534 024171         LDA 1,C11
003535 153103         ADDL 2,2,SNC
003536 024117         LDA 1,C3          3
003537 123000         ADD 1,0
003540 063033         DOC 0,DSKP      ;LOAD UNIT # & # SECTORS
003541 002067         LDA 0,KB7         WRITE
003542 061133         DOAS 0,DSKP     ;WRITE !!
003543 152520         SUBZL 2,2
003544 006227         WAIT            ;WAIT 100 MS (OR UNTIL "DONE")
003545 066433         DIC 1,DSKP      ;READ ENDING DISK ADDRESS
003546 003133         LDA 2,C037
003547 147400         AND 2,1         ;THROW AWAY UNIT #                  420
003550 002212         LDA 0,C420      ;ACO=GOOD,    HEAD-1 SECT-1         417
003551 122414         SUB# 1,0,SZR    ;AC1=BAD
003552 006242         EHALT           ;CHECK "ADV HD", "INC HEAD"
003553 006243         LOOP

003554 006240 D30:    SETP1           ;ATTEMPT A 16 SECTOR WRITE
003555 002160         LDA 0,BUFF      ;VERIFY CORRECT ENDING
003556 062033         DOB 0,DSKP      ;DISK ADDRESS
003557 002152         LDA 0,TESTII
003560 063033         DOC 0,DSKP      ;UNIT # / 16 SECTORS
003561 002067         LDA 0,KB7
003562 061133         DOAS 0,DSKP     ;WRITE !!
003563 152520         SUBZL 2,2
003564 006227         WAIT            ;WAIT 100MS (OR UNTIL "DONE")
003565 066433         DIC 1,DSKP      ;READ ENDING DISK ADDRESS
003566 002065         LDA 0,KB9       ;SECTOR 4
003567 034070         LDA 3,KB6       ;HEAD 2
003570 003150         LDA 2,DTYPE
003571 153103         ADDL 2,2,SNC    ;SKIP IF 2311
003572 175220         MOVZR 3,3       ;HEAD 1
003573 163000         ADD 3,0
003574 034133         LDA 3,C037
003575 167400         AND 3,1
003576 122414         SUB# 1,0,SZR    ;AC1=ACTUAL ENDING DISK ADDRESS
003577 006242         EHALT           ;AC0=CORRECT ENDING DISK ADDRESS
003600 006243         LOOP            ; 16 SECTOR TRANSFER.
```

```
A 0046  .MAIN

003601 006240 D31:    SETP1             ;ATTEMPT A 16 SECTOR WRITE
003602 002160         LDA 0,BUFF        ;AND VERIFY CORRECT ENDING
003603 062033         DOR 0,DSKP        ;MEMORY ADDRESS
003604 002152         LDA 0,TESTU
003605 063033         DOC 0,DSKP        ;UNIT # / 16 SECTORS
003606 002067         LDA 0,KB7
003607 061133         DOAS 0,DSKP       ;WRITE !!
003610 152520         SUBZL 2,2
003611 006227         WAIT              ;WAIT 100MS (OR UNTIL "DONE")
003612 002160         LDA 0,BUFF        ;STARTING ADDRESS
003613 024000         LDA 1,KB14        ; TWO
003614 030073         LDA 2,KB3         ; 10,000
003615 123000         ADD 1,0
003616 143000         ADD 2,0           ;ENDING MEMORY ADDRESS
003617 065433         DIB 1,DSKP        ;ERROR FOLLOWING A 16 SECTOR -
003620 122414         SUB# 1,0,SZR      ;WRITE
003621 006242         EHALT             ;AC0=GOOD
003622 006243         LOOP              ;AC1=BAD

003623 006240 D32:    SETP1             ;CAUSE "END OF CYLINDER"
003624 002150         LDA 0,DTYPE       ;TO OCCUR DURING A
003625 030427         LDA 2,DAD1-1      ;2 SECTOR WRITE
003626 101112         MOVL# 0,0,SZC
003627 000004         JMP .+4           ;CARTRIDGE DISK
003630 151400         INC 2,2
003631 101202         MOVR 0,0,SZC      ;SKIP IF 2311 DISK PACK
003632 151000         INC 2,2           ;2314 DISK PACK
003633 021000         LDA 0,0,2         ;GET PROPER DISK ADDRESS TO
003634 024152         LDA 1,TESTU       ;CAUSE "EOC"
003635 123000         ADD 1,0           ;ADD IN UNIT #
003636 063033         DOC 0,DSKP        ;FINALLY
003637 002160         LDA 0,BUFF
003640 062033         DOR 0,DSKP        ;LOAD MEM ADDR.
003641 002067         LDA 0,KB7
003642 061133         DOAS 0,DSKP       ;WRITE !!
003643 152520         SUBZL 2,2
003644 006227         WAIT              ;WAIT 100MS (OR UNTIL "DONE")
003645 064433         DIA 1,DSKP        ;READ STATUS
003646 020412         LDA 0,DAD4
003647 107400         AND 0,1
003650 100414         SUB# 0,1,SZR      ;NO "EOC" OR "ERR" OR "DP DONE"
003651 006242         EHALT             ;FOLLOWING WRITE OVER END CYL
003652 006243         LOOP              ;AC0=EXPECTED ENDING STATUS
003653 000446         JMP D33           ;AC1=ENDING STATUS

003654 003655         .+1
003655 000676 DAD1:   676               ;CART   HEAD 1  SECT 13, 2 SECTORS
003656 004536         4536              ;2311   HEAD 9  SECT 5  2 SECTORS
003657 011676         11676             ;2314   HEAD 23 SECT 13 2 SECTORS
003660 100021 DAD4:   100021            ;"DP DONE","EOC","ERR",STATUS MASK.
```

```
03661 006240 D33:   SETP1              ;ATTEMPT A READ
03662 020152        LDA 0,TESTU        ;( FIRST READ !!!!! )
03663 024121        LDA 1,C17
03664 123000        ADD 1,0
03665 063033        DOC 0,DSKP         ;UNIT # / 1 SECTOR XFER
03666 020074        LDA 0,KB2
03667 062033        DOB 0,DSKP         ;CA = 20000
03670 102400        SUB 0,0
03671 061133        DOAS 0,DSKP        ;READ !!
03672 152520        SUBZL 2,2
03673 006227        WAIT               ;WAIT 100MS (OR UNTIL "DONE")
03674 064433        DIA 1,DSKP         ;READ STATUS
03675 125113        MOVL# 1,1,SNC      ; DONE ?
03676 006242        FHALT              ;NO "OP DONE" ON READ
03677 006243        LOOP               ;AC1=BAD ENDING STATUS

03700 006240 D34:   SETP1              ;ATTEMPT A READ
03701 020152        LDA 0,TESTU        ; 1 SECTOR
03702 024121        LDA 1,C17
03703 123000        ADD 1,0
03704 063033        DOC 0,DSKP         ;UNIT # / 1 SECTOR XFER
03705 020074        LDA 0,KB2
03706 062033        DOB 0,DSKP         ;CA = 20000
03707 102400        SUB 0,0
03710 061133        DOAS 0,DSKP        ;READ !!
03711 152520        SUBZL 2,2
03712 006227        WAIT               ;WAIT 100MS (OR UNTIL "DONE")
03713 064433        DIA 1,DSKP         ;READ STATUS
03714 020060        LDA 0,KB14         ;"DATA LATE" STATUS ON
03715 123414        AND# 1,0,SZR       ;A 1 SECT READ
03716 006242        FHALT              ;AC1=BAD ENDING STATUS
03717 006243        LOOP

03720 006240 D35:   SETP1              ;ATTEMPT A READ
03721 020152        LDA 0,TESTU        ; 1 SECTOR READ
03722 024121        LDA 1,C17
03723 123000        ADD 1,0
03724 063033        DOC 0,DSKP         ; UNIT # / 1 SECT XFER
03725 020074        LDA 0,KB2
03726 062033        DOB 0,DSKP         ;CA = 20000
03727 102400        SUB 0,0
03730 061133        DOAS 0,DSKP        ;READ !!
03731 152520        SUBZL 2,2
03732 006227        WAIT               ;WAIT 100MS (OR UNTIL "DONE")
03733 065433        DIB 1,DSKP         ;READ STATUS
03734 020217        LDA 0,C204H        ;ENDING MEMORY ADDRESS
03735 122414        SUB# 1,0,SZR       ;IS WRONG FOLLWOING 1 SECT READ
03736 006242        FHALT              ;AC0=GOOD
03737 006243        LOOP               ;AC1=BAD
```

20400
20753

```
▲ 0004  .MAIN

0374A 006240 D36:    SETP1              ;ATTEMPT A READ
03741 002A152        LDA A,TESTU        ; 1 SECTOR READ
03742 024121         LDA 1,C17
03743 123000         ADD 1,0
03744 063033         DOC 0,DSKP         ;UNIT # / 1 SECT XFER
03745 002074         LDA 0,KB2
03746 062033         DOA 0,DSKP         ;CA = 20000
03747 060133         NIOS DSKP          ;READ !!
03750 152520         SUBZL 2,2
03751 006227         WAIT               ;WAIT 100MS (OR UNTIL "DONE")
03752 002002         LDA 0,C70          ;AC1=STATUS
03753 123414         AND# 1,0,SZR       ;ERROR STATUS FOLLOWING A
03754 006242         FHALT              ;READ, "SEEK ER", OR "ADDRESS/
03755 006243         LOOP               ;UNSAFE"

03756 002A154 D37:   LDA A,DTYPE        ;IF CARTRIDGE DISK
03757 101102         MOVL A,A,SZC       ;DON'T ATTEMPT A FORMAT
03760 000416         JMP E1             ;MODE READ
03761 006240         SETP1              ;ATTEMPT A READ
03762 002A152        LDA A,TESTU        ;IN THE FORMAT MODE
03763 024074         LDA 1,KB2          ;(FIRST USE OF FORMAT MODE)
03764 123000         ADD 1,0
03765 063033         DOC A,DSKP         ;UNIT # FORMAT MODE BIT
03766 002074         LDA A,KB2
03767 062033         DOA A,DSKP         ;CA = 20000
03770 060133         NIOS DSKP          ;READ!!
03771 152520         SUBZL 2,2
03772 006227         WAIT               ;WAIT 100MS (OR UNTIL "DONE")
03773 125113         MOVL# 1,1,SNC      ;AC1=STATUS
03774 006242         FHALT              ;NO "DP DONE" FOLLOWING FMT READ
03775 006243         LOOP               ;CHECK "F DONE","R/W DONE"

                     .EOT
```

```
0049  .MAIN

003776 006237 E1:     JSR @ISET          ;DO SEEK/WRITE/READ
003777 006262         DOSEK              ;CHECK DATA
004000 000000         0                  ;SEEK CYLINDER ZERO
004001 006242         EHALT              ;ERROR DURING SEEK, AC1=STATUS
004002 000422         JMP E1E            ;SKIP REMAINDER OF TEST

004003 006256         GENDAT             ;GENERATE DATA
004004 005403         ZEROS              ;DATA=ALL ZEROS
004005 006605         PRGEND             ;DATA BUFFER ADDRESS

004006 006260         WRITE              ;DO THE WRITE
004007 006605         PRGEND             ;DATA BUFFER ADDRESS
004010 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
004011 006242         EHALT              ;ERROR IN WRITE, AC1=STATUS
004012 000412         JMP E1E            ;SKIP REMAINDER OF TEST

004013 006257         READ               ;READ THE DATA
004014 007205         PRGEND+400         ;DATA BUFFER ADDRESS
004015 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
004016 006242         EHALT              ;ERROR IN READ, AC1=STATUS
004017 000405         JMP E1E

004020 006261         CHECK              ;COMPAPE DATA BUFFERS
004021 006605         PRGEND             ;GOOD BUFFER
004022 007205         PRGEND+400         ;QUESTIONABLE BUFFER
004023 006242         EHALT              ;ERROR, AC0=GOOD WORD
004024 006243 E1E:    LOOP               ;AC1=BAD WORD

004025 006237 E2:     JSR @ISET          ;DO SEEK/WRITE/READ
004026 006262         DOSEK              ;CHECK DATA
004027 000000         0                  ;SEEK CYLINDER ZERO
004030 006242         EHALT              ;ERROR DURING SEEK, AC1=STATUS
004031 000422         JMP E2E            ;SKIP REMAINDER OF TEST

004032 006256         GENDAT             ;GENERATE DATA
004033 005402         ONES               ;DATA=ALL ONES
004034 006605         PRGEND             ;DATA BUFFER ADDRESS

004035 006260         WRITE              ;DO THE WRITE
004036 006605         PRGEND             ;DATA BUFFER ADDRESS
004037 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
004040 006242         EHALT              ;ERROR IN WRITE, AC1=STATUS
004041 000412         JMP E2E            ;SKIP REMAINDER OF TEST

004042 006257         READ               ;READ THE DATA
004043 007205         PRGEND+400         ;DATA BUFFER ADDRESS
004044 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
004045 006242         EHALT              ;ERROR IN READ, AC1=STATUS
004046 000405         JMP E2F

004047 006261         CHECK              ;COMPARE DATA BUFFERS
004050 006605         PRGEND             ;GOOD BUFFER
004051 007205         PRGEND+400         ;QUESTIONABLE BUFFER
004052 006242         EHALT              ;ERROR, AC0=GOOD WORD
004053 006243 E2E:    LOOP               ;AC1=BAD WORD
```

```
04054 006237 E3:      JSR @ISET          ;DO SEEK/WRITE/READ
04055 006262          DOSEK              ;CHECK DATA
04056 000000          0                  ;SEEK CYLINDER ZERO
04057 006242          FHALT              ;ERROR DURING SEEK, AC1=STATUS
04060 000422          JMP E3E            ;SKIP REMAINDER OF TEST

04061 006256          GENDAT             ;GENERATE DATA
04062 005421          NUMSEQ             ;DATA=NUMBERS 0-377
04063 006605          PRGEND             ;DATA BUFFER ADDRESS

04064 006260          WRITE              ;DO THE WRITE
04065 006605          PRGEND             ;DATA BUFFER ADDRESS
04066 000017          17                 ;HEAD=0, SECT=0, 1 SECTOR
04067 006242          FHALT              ;ERROR IN WRITE, AC1=STATUS
04070 000412          JMP E3E            ;SKIP REMAINDER OF TEST

04071 006257          READ               ;READ THE DATA
04072 007205          PRGEND+400         ;DATA BUFFER ADDRESS
04073 000017          17                 ;HEAD=0, SECT=0, 1 SECTOR
04074 006242          FHALT              ;ERROR IN READ, AC1=STATUS
04075 000405          JMP E3E

04076 006261          CHECK              ;COMPARE DATA BUFFERS
04077 006605          PRGEND             ;GOOD BUFFER
04100 007205          PRGEND+400         ;QUESTIONABLE BUFFER
04101 006242          FHALT              ;ERROR, AC0=GOOD WORD
04102 006243 E3E:     LOOP               ;AC1=BAD WORD

04103 006237 E4:      JSR @ISET          ;DO SEEK/WRITE/READ
04104 006262          DOSEK              ;CHECK DATA
04105 000000          0                  ;SEEK CYLINDER ZERO
04106 006242          FHALT              ;ERROR DURING SEEK, AC1=STATUS
04107 000422          JMP E4E            ;SKIP REMAINDER OF TEST

04110 006256          GENDAT             ;GENERATE DATA
04111 005411          ALT1               ;DATA PATTERN = 1010101 ETC.
04112 006605          PRGEND             ;DATA BUFFER ADDRESS

04113 006260          WRITE              ;DO THE WRITE
04114 006605          PRGEND             ;DATA BUFFER ADDRESS
04115 000017          17                 ;HEAD=0, SECT=0, 1 SECTOR
04116 006242          FHALT              ;ERROR IN WRITE, AC1=STATUS
04117 000412          JMP E4E            ;SKIP REMAINDER OF TEST

04120 006257          READ               ;READ THE DATA
04121 007205          PRGEND+400         ;DATA BUFFER ADDRESS
04122 000017          17                 ;HEAD=0, SECT=0, 1 SECTOR
04123 006242          FHALT              ;ERROR IN READ, AC1=STATUS
04124 000405          JMP E4E

04125 006261          CHECK              ;COMPARE DATA BUFFERS
04126 006605          PRGEND             ;GOOD BUFFER
04127 007205          PRGEND+400         ;QUESTIONABLE BUFFER
04130 006242          FHALT              ;ERROR, AC0=GOOD WORD
04131 006243 E4E:     LOOP               ;AC1=BAD WORD
```

```
A 0051  .MAIN

 04132 006237 E5:    JSR @ISET        ;DO SEEK/WRITE/READ
 04133 006262        DOSEK            ;CHECK DATA
 04134 000000        0                ;SEEK CYLINDER ZERO
 04135 006242        EHALT            ;ERROR DURING SEEK, AC1=STATUS
 04136 000422        JMP E5E          ;SKIP REMAINDER OF TEST

 04137 006256        GENDAT           ;GENERATE DATA
 04140 005413        ALTW             ;DATA PATTERN = 0101010 ETC.
 04141 006605        PRGEND           ;DATA BUFFER ADDRESS

 04142 006260        WRITE            ;DO THE WRITE
 04143 006605        PRGEND           ;DATA BUFFER ADDRESS
 04144 000017        17               ;HEAD=0, SECT=0, 1 SECTOR
 04145 006242        EHALT            ;ERROR IN WRITE, AC1=STATUS
 04146 000412        JMP E5F          ;SKIP REMAINDER OF TEST

 04147 006257        READ             ;READ THE DATA
 04150 007205        PRGEND+400       ;DATA BUFFER ADDRESS
 04151 000017        17               ;HEAD=0, SECT=0, 1 SECTOR
 04152 006242        EHALT            ;ERROR IN READ, AC1=STATUS
 04153 000405        JMP E5F

 04154 006261        CHECK            ;COMPARE DATA BUFFERS
 04155 006605        PRGEND           ;GOOD BUFFER
 04156 007205        PRGEND+400       ;QUESTIONABLE BUFFER
 04157 006242        EHALT            ;ERROR, AC0=GOOD WORD
 04160 006243 E5E:   LOOP             ;AC1=BAD WORD
```

```
▲ 0052  .MAIN

 04161 020142          LDA 0,RANDOM
 04162 040143          STA 0,RELRAN
 04163 006237 E6:      JSR @ISET       ;DO SEEK/WRITE/READ
 04164 006262          DOSEK           ;CHECK DATA
 04165 000000          0               ;SEEK CYLINDER ZERO
 04166 006242          EHALT           ;ERROR DURING SEEK, AC1=STATUS
 04167 000424          JMP E6E         ;SKIP REMAINDER OF TEST

 04170 020143          LDA 0,RELRAN
 04171 040142          STA 0,RANDOM
 04172 006256          GENDAT          ;GENERATE DATA
 04173 005217          RAN             ;DATA = RANDOM
 04174 006605          PRGEND          ;DATA BUFFER ADDRESS

 04175 006260          WRITE           ;DO THE WRITE
 04176 006605          PRGEND          ;DATA BUFFER ADDRESS
 04177 000017          17              ;HEAD=0, SECT=0, 1 SECTOR
 04200 006242          EHALT           ;ERROR IN WRITE, AC1=STATUS
 04201 000412          JMP E6E         ;SKIP REMAINDER OF TEST

 04202 006257          READ            ;READ THE DATA
 04203 007205          PRGEND+400      ;DATA BUFFER ADDRESS
 04204 000017          17              ;HEAD=0, SECT=0, 1 SECTOR
 04205 006242          EHALT           ;ERROR IN READ, AC1=STATUS
 04206 000405          JMP E6E

 04207 006261          CHECK           ;COMPARE DATA BUFFERS
 04210 006605          PRGEND          ;GOOD BUFFER
 04211 007205          PRGEND+400      ;QUESTIONABLE BUFFER
 04212 006242          EHALT           ;ERROR, AC0=GOOD WORD
 04213 006243 E6E:     LOOP            ;AC1=BAD WORD
```

```
;DISCUSSION OF TESTS E7/E8 AND E9/E10

;       AT THIS POINT IN TESTING IT HAS BEEN DETERMINED
;THAT READING AND WRITING CAN BE PERFORMED CORRECTLY.  THE
;NEXT FOUR TESTS ARE A CYLINDER ADDRESS CHECK.  CYLINDERS
;ARE FIRST SELECTED IN ORDER (0-312) AND DATA EQUAL TO THE
;CYLINDER NUMBER IS WRITTEN (TEST E7) ON HEAD-0
;SECTOR-0 OF EACH.  THE FOLLOWING TEST THEN READS THIS DATA
;BACK IN THE SAME SEQUENCE IN WHICH IT WAS WRITTEN.  IF ONE
;CYLINDER IS WRITTEN ON MORE THAN ONCE DUE TO A (ALWAYS 0 OR
;ALWAYS 1) CYLINDER ADDRESS BIT LINE AN ERROR WILL SHOW UP
;IN THE DATA COMPARE CHECK.

; I.E.  IF "CYL 4" IS ALWAYS A ZERO THEN AN ATTEMPT
;       TO SEEK TO CYLINDER 4 ACTUALLY SELECTS
;       CYLINDER 0.  DATA WORDS OF "4" ARE WRITTEN
;       THERE AND WHEN THE DATA AT CYLINDER 0 IS
;       LATER READ AND CHECKED IT WILL BE "4" WHEN
;       IT SHOULD BE "0".

;TESTS E7&E8 CHECK ALL CYLINDERS IN SEQUENCE 0-312
;AND TESTS E9&E10 CHECK THE CYLINDERS IN ORDER
;FROM 312 TO 0, USING THE COMPLEMENT OF THE CYLINDER
;NUMBER AS THE DATA WORDS.
```

```
A 0054  .MAIN

 04214 102400           SUB 0,0          ;CYL. ADDRESSING CHECK. SEE DISCUSSION
 04215 040403           STA 0,E7.1       ;PRECEDING E7. WRITE ON HEAD=0, SECTOR=0
 04216 006237 E7:       JSR @ISET        ;OF EACH CYLINDER.  IN EACH SECTOR DATA
                                         ;EQUALS THE CYL #.  WRITE IN ASCENDING
                                         ;ORDER FROM CYL 0 TO 312.
 04217 006262           DOSEK            ;SEEK
 04220 000000 E7.1:     0                ;CYLINDER NUMBER (IT CHANGES)
 04221 006242           FHALT            ;ERROR DURING SEEK, AC1=STATUS
 04222 000411           JMP E7E          ;SKIP TO END OF TEST
 04223 006256           GENDAT           ;GENERATE DATA
 04224 005417           CYLN             ;ADDRESS OF DATA GENERATOR
 04225 006605           PRGEND           ;DATA BUFFER ADDRESS
 04226 006260           WRITE            ;WRITE !
 04227 006605           PRGEND           ;DATA BUFFER ADDRESS
 04230 000017           17               ;DISK ADDRESS
 04231 006242           FHALT            ;ERROR DURING WRITE, AC1=STATUS
 04232 000401           JMP .+1
 04233 006243 E7F:      LOOP
 04234 010764           ISZ E7.1         ;(E7.1)=CYL #
 04235 020763           LDA 0,E7.1       ;DO IT ONCE FOR
 04236 024207           LDA 1,C312       ;EACH CYLINDER
 04237 122427           SUBZ 1,0,SBN
 04240 000756           JMP E7



 04241 102400           SUB 0,0          ;CYLINDER ADDRESS CHECK.  SEE DISCUSSION
 04242 040403           STA 0,E8.1       ;PREVIOUS TO TEST E7.  READ FIRST SECTOR
 04243 006237 E8:       JSR @ISET        ;OF EACH CYLINDER AND CHECK FOR PROPER
                                         ;DATA.  IN EACH CASE DATA SHOULD EQUAL
                                         ;THE CYLINDER #.
 04244 006262           DOSEK            ;SEEK
 04245 000000 E8.1:     0                ;CYLINDER NUMBER (IT CHANGES)
 04246 006242           FHALT            ;ERROR DURING SEEK, AC1=STATUS
 04247 000412           JMP E8E
 04250 006257           READ             ;READ
 04251 006605 E8.2:     PRGEND           ;DATA BUFFER ADDRESS
 04252 000017           17               ;DISK ADDRESS
 04253 006242           FHALT            ;ERROR DURING READ, AC1=STATUS
 04254 000405           JMP E8F          ;SKIP TO END OF TEST
 04255 024774           LDA 1,@E8.2      ;FIRST WORD IN DATA BUFFER
 04256 020767           LDA 0,E8.1       ;SHOULD = CYLINDER #
 04257 122414           SUB# 1,0,SZR     ;AC0=GOOD
 04260 006242           FHALT            ;AC1=BAD
 04261 006243 E8F:      LOOP             ;READ THE TEST DESCRIPTION
 04262 010763           ISZ E8.1         ;REPEAT THE TEST FOR
 04263 020762           LDA 0,E8.1       ;EACH CYLINDER
 04264 024207           LDA 1,C312
 04265 122427           SUBZ 1,0,SBN
 04266 000755           JMP E8
```

```
04267 020207          LDA 0,C312      ;CYL. ADDRESSING CHECK. SEE DISCUSSION
04270 040403          STA 0,E9.1      ;PRECEDING TEST E7.  WRITE ON HEAD-0
04271 006237 E9:      JSR @ISET       ;SECTOR-0, OF EACH CYLINDER.  IN
                                      ;EACH THE DATA WORDS EQUAL THE COMP.
                                      ;OF THE CYL #.  WRITE IN DESCENDING
                                      ;ORDER FROM CYL 312 TO 0.
04272 006262          DOSEK           ;SEEK
04273 000000 E9.1:    0               ;CYLINDER NUMBER (IT CHANGES)
04274 006242          EHALT           ;ERROR DURING SEEK, AC1=STATUS
04275 000411          JMP E9E         ;SKIP TO END OF TEST
04276 006256          GENDAT          ;GENERATE DATA
04277 005415          CYLNC           ;ADDRESS OF DATA GENERATOR
04300 006605          PRGEND          ;DATA BUFFER ADDRESS
04301 006260          WRITE           ;WRITE
04302 006605          PRGEND          ;DATA BUFFER ADDRESS
04303 000017          17              ;DISK ADDRESS
04304 006242          EHALT           ;ERROR DURING WRITE, AC1=STATUS
04305 000401          JMP .+1
04306 006243 E9E:     LOOP
04307 014764          DSZ E9.1        ;(E9.1)=CYL #
04310 000401          JMP .+1
04311 020762          LDA 0,E9.1      ;DO IT ONCE FOR
04312 101103          MOVL 0,0,SNC    ;EACH CYLINDER
04313 000756          JMP E9


04314 020207          LDA 0,C312      ;CYLINDER ADDRESS CHECK.  SEE DISCUSSIO
04315 040403          STA 0,E10.1     ;PREVIOUS TO TEST E7.  READ FIRST SECTO
04316 006237 E10:     JSR @ISET       ;OF EACH CYLINDER AND CHECK FOR PROPER
                                      ;DATA.  IN EACH CASE DATA SHOULD EQUAL
                                      ;THE COMPLEMENT OF THE CYL #
04317 006262          DOSEK           ;SEEK
04320 000000 E10.1:   0               ;CYLINDER NUMBER (IT CHANGES)
04321 006242          EHALT           ;ERROR DURING SEEK, AC1=STATUS
04322 000413          JMP E10E
04323 006257          READ            ;READ
04324 006605 E10.2:   PRGEND          ;DATA BUFFER ADDRESS
04325 000017          17              ;DISK ADDRESS
04326 006242          EHALT           ;ERROR DURING READ, AC1=STATUS
04327 000406          JMP E10E        ;SKIP TO END OF TEST
04330 026774          LDA 1,@E10.2    ;FIRST WORD IN DATA BUFFER
04331 020767          LDA 0,E10.1     ;SHOULD = CYLINDER #
04332 100000          COM 0,0
04333 122414          SUB# 1,0,SZR    ;AC0=GOOD
04334 006242          EHALT           ;AC1=BAD
04335 006243 E10E:    LOOP            ;READ THE TEST DESCRIPTION
04336 014762          DSZ E10.1       ;REPEAT THE TEST FOR
04337 000401          JMP .+1
04340 020760          LDA 0,E10.1     ;EACH CYLINDER
04341 101103          MOVL 0,0,SNC
04342 000754          JMP E10

                      .EOT
```

```
                ;DISCUSSION OF THE SECTOR ADDRESSING CHECK.
                ;(TESTS E11/E12 AND E13/E14)

                ;USING CYLINDER-0, HEAD-0 TEST E11 WRITES ON
                ;EACH SUCCESSIVE SECTOR INDIVIDUALLY FROM 0-5,
                ;OR 0-11.  IN EACH CASE EACH DATA WORD EQUALS
                ;THE SECTOR NUMBER.  TEST E12 READS THE INDIV-
                ;IDUAL SECTORS BACK IN THE SAME ORDER AND
                ;CHECKS THE DATA.  TESTS E13/E14 PERFORM THE SAME
                ;TASK EXCEPT THAT THE SECTOR SEQUENCE IS IN THE
                ;REVERSE ORDER AND THE DATA WRITTEN EQUALS THE
                ;COMPLEMENT OF THE SECTOR NUMBER.

                ;THIS TEST IS DESIGNED TO CATCH ERRONEOUS SECTOR
                ;SELECTION ERRORS.

                ;  I.E. IF "SC2" IS ALWAYS AT GROUND WHEN SECTOR
                ;       2 IS SELECTED, SECTOR 0 WILL ACTUALLY
                ;       BE CHOSEN AND "2'S" WILL BE WRITTEN THERE.
                ;       IN THE SUBSEQUENT READ A DATA COMPARE
                ;       ERROR WILL RESULT WITH THE GOOD="0",
                ;       AND THE BAD="2".

004343 102400           SUB 0,0             ;SEE DISCUSSION PRECEDING THIS TEST.
004344 040157           STA 0,SECT          ;WRITE ON CYL-0, HEAD-0, ONCE
004345 020121           LDA 0,C17           ;FOR EACH SECTOR.  IN EACH
004346 040413           STA 0,E11.1         ;CASE THE DATA WORDS EQUAL
004347 006237 F11:      JSR @ISET           ;THE SECTOR NUMBER
004350 006262           DOSEK               ;SEEK !
004351 000000           0                   ;CYLINDER 0
004352 006242           EHALT               ;ERROR IN SEEK, AC1=STATUS
004353 000411           JMP E11E            ;SKIP TO END OF TEST
004354 006256           GENDAT              ;GENERATE DATA
004355 005440           SECTN               ;ADDRESS OF DATA GEN ROUT.
004356 006605           PRGEND              ;ADDRESS OF DATA BUFFER
004357 006250           WRITE               ;WRITE !.
004360 006605           PRGEND              ;DATA BUFFER ADDRESS
004361 000017 F11.1:    17                  ;DISK ADDRESS (IT CHANGES)
004362 006242           EHALT               ;ERROR IN WRITE, AC1=STATUS
004363 000401           JMP .+1
004364 006243 E11E:     LOOP

004365 040157           ISZ SECT            ;INCREMENT THE SECTOR
004366 023773           LDA 0,F11.1         ;NUMBER
004367 024053           LDA 1,K011
004370 123000           ADD 1,0
004371 043773           STA 0,E11.1
004372 024150           LDA 1,DTYPE         ;SEE IF DONE
004373 034205           LDA 2,C157
004374 127103           ADDL 1,1,SNC
004375 030210           LDA 2,C317
004376 112404           SUB 1,2,SZR
004377 000750           JMP F11             ;DO ANOTHER
```

```
▲ 0057  .MAIN

 004400 102400          SUB 0,0          ;SEE THE DISCUSSION PRECEDING
 004401 040157          STA 0,SECT       ;TEST E11.  READ ONCE FROM
 004402 020121          LDA 0,C17        ;EACH SECTOR AT CYL=0, HEAD=0.
 004403 040410          STA 0,E12.2      ;CHECK DATA IN EACH CASE.
 004404 006237 E12:     JSR @ISET        ;DATA WORDS=SECTOR #.
 004405 006262          DOSEK            ;SEEK
 004406 000000          0                ;CYLINDER 0
 004407 006242          EHALT            ;ERROR IN SEEK, AC1=STATUS
 004410 000412          JMP E12E         ;SKIP TO END OF TEST
 004411 006257          READ             ;READ !
 004412 006605 E12.1:   PRGEND           ;DATA BUFFER ADDRESS
 004413 000017 E12.2:   17               ;DISK ADDRESS (IT CHANGES)
 004414 006242          EHALT            ;ERROR IN READ, AC1=STATUS ←
 004415 000405          JMP E12E         ;SKIP TO END OF TEST
 004416 020157          LDA 0,SECT       ;GET SECT #
 004417 026773          LDA 1,@E12.1     ;GET A WORD READ
 004420 122414          SUB# 1,0,SZR     ;DATA ERROR, SEE ABOVE DESCRIPTION
 004421 006242          EHALT            ;AC0=GOOD WORD
 004422 006243 E12E:    LOOP             ;AC1=BAD

 004423 010157          ISZ SECT         ;INCREMENT THE SECTOR
 004424 020767          LDA 0,E12.2      ;NUMBER
 004425 024063          LDA 1,K811
 004426 123000          ADD 1,0
 004427 040764          STA 0,E12.2
 004430 024150          LDA 1,DTYPE      ;SEE IF DONE
 004431 030205          LDA 2,C157
 004432 127103          ADDL 1,1,SNC
 004433 030210          LDA 2,C317
 004434 112404          SUB 0,2,SZR
 004435 000747          JMP E12          ;GO AGAIN
```

```
04436 020150          LDA 0,DTYPE        ;SEE DISCUSSION PRECEDING
04437 024204          LDA 1,C137         ;TEST E11.  WRITE ON CYL-0,
04440 103103          ADDL 0,0,SNC       ;HEAD-0, ONCE FOR EACH SECTOR.
04441 024206          LDA 1,C277         ;IN EACH CASE THE DATA EQUALS
04442 044420          STA 1,E13.1        ;THE COMPLEMENT OF THE SECTOR
04443 125220          MOVZR 1,1          ;NUMBER
04444 125220          MOVZR 1,1
04445 125220          MOVZR 1,1
04446 125220          MOVZR 1,1
04447 044157          STA 1,SECT         ;BEGINNING SECTOR
04450 006237 E13:     JSR @ISET          ;SETUP
04451 006262          DOSEK              ;SEEK !
04452 000000          0                  ;CYLINDER 0
04453 006242          EHALT              ;ERROR IN SEEK, AC1=STATUS
04454 000411          JMP E13E           ;SKIP TO END OF TEST
04455 006256          GENDAT             ;GENERATE DATA
04456 005436          SETNC              ;ADDRESS OF DATA GENERATOR
04457 006605          PRGEND             ;ADDRESS OF DATA BUFFER
04460 006260          WRITE              ;WRITE !
04461 006605          PRGEND             ;ADDRESS OF DATA BUFFER
04462 000017 E13.1:   17                 ;DISK ADDRESS (IT CHANGES)
04463 006242          EHALT              ;ERROR IN WRITE, AC1=STATUS
04464 000401          JMP .+1
04465 006243 E13E:    LOOP

04466 014157          DSZ SECT           ;DECREMENT THE SECTOR
04467 000401          JMP .+1            ;NUMBER
04470 020772          LDA 0,E13.1
04471 024063          LDA 1,KB11
04472 122400          SUB 1,0
04473 042767          STA 0,E13.1
04474 101404          INC 0,0,SZR        ;SEE IF DONE
04475 000753          JMP E13            ;NOT YET
```

```
004476 020150          LDA 0,DTYPE      ;SEE THE DISCUSSION PRECEDING
004477 024004          LDA 1,C137       ;TEST 11.  READ ONCE FROM
004500 103403          ADDL 0,0,SNC     ;SECTOR OF CYL-0, HEAD-0,
004501 024006          LDA 1,C277       ;READ SUCCESSIVE SECTORS
004502 044415          STA 1,E14.2      ;IN ORDER FROM HI TO LOW.
004503 125220          MOVZR 1,1        ;IN EACH CASE THE DATA EQUALS
004504 125220          MOVZR 1,1        ;THE COMPLEMENT OF THE
004505 125220          MOVZR 1,1        ;SECTOR NUMBER
004506 125220          MOVZR 1,1
004507 044157          STA 1,SECT       ;STARTING SECTOR #
004510 006237 F14:     JSR @ISET
004511 006262          DOSEK            ;SEEK !
004512 000000          0                ;CYLINDER 0
004513 006242          FHALT            ;ERROR IN SEEK, AC1=STATUS
004514 000413          JMP E14E         ;SKIP TO END OF TEST
004515 006257          READ             ;READ !
004516 006605 E14.1:   PRGEND           ;DATA BUFFER ADDRESS
004517 000017 E14.2:   17               ;DISK ADDRESS (IT CHANGES)
004520 006242          FHALT            ;ERROR IN READ, AC1=STATUS
004521 000406          JMP E14E         ;SKIP TO END OF TEST
004522 020157          LDA 0,SECT       ;GET SECTOR #
004523 100000          COM 0,0
004524 026772          LDA 1,@E14.1     ;GET A WORD READ
004525 122414          SUB# 1,0,SZR     ;DATA ERROR, SEE ABOVE DESCRIPTION
004526 006242          FHALT            ;AC0=GOOD WORD
004527 006243 E14E:    LOOP             ;AC1=BAD

004530 014157          DSZ SECT         ;DECREMENT TO NEXT SECTOR
004531 000401          JMP .+1
004532 020765          LDA 0,E14.2
004533 024063          LDA 1,KB11
004534 122400          SUB 1,0
004535 040762          STA 0,E14.2
004536 101404          INC 0,0,SZR      ;DONE ?
004537 000751          JMP E14          ;NO, GO AGAIN
```

```
;DISCUSSION OF THE HEAD ADDRESSING CHECK.
; (TESTS E15/E16 AND E17/E18)

;USING CYLINDER-0, SECTOR-0, TEST E15 WRITES ONE
;SECTOR ON EACH SUCCESSIVE HEAD INDIVIDUALLY.
;IN EACH CASE THE DATA WORDS EQUAL THE HEAD
;NUMBER.  TEST E16 READS EACH OF THESE SECTORS
;BACK IN THE SAME ORDER THEY WERE WRITTEN AND
;CHECKS THE DATA.  TESTS E17 AND E18 PERFORM
;THE SAME FUNCTIONS EXCEPT THAT THE HEADS ARE
;SELECTED IN REVERSE ORDER AND THE COMPLEMENT
;OF THE HEAD NUMBER IS USED AS THE DATA.

;THESE TEST ARE DESIGNED TO CATCH ERRONEOUS
;HEAD SELECTION OR MULTIPLE HEAD SELECTION
;ERRORS.

;  I.E. IF "HD2" IS ALWAYS AT GROUND WHEN HEAD 2
;       IS SELECTED, HEAD 0 WILL ACTUALLY BE
;       CHOSEN.  "2'S" WILL BE WRITTEN ON THIS
;       SECTOR WHERE "0'S" HAD BEEN WRITTEN
;       PREVIOUSLY.  THE SUBSEQUENT READ WILL
;       THEN ENCOUNTER A DATA ERROR WITH THE
;       GOOD WORD = 0 AND THE BAD WORD = 2.
```

```
▲ 00061  .MAIN

 04540 102400          SUB 0,0          ;SEE THE DISCUSSION ABOVE.
 04541 040156          STA 0,HEAD       ;WRITE ON CYL-0, SECT-0,
 04542 020121          LDA 0,C17        ;ONCE FOR EACH HEAD.
 04543 040413          STA 0,E15.1      ;DATA WORDS = HEAD #
 04544 006237 E15:     JSR @ISET
 04545 006262          DOSEK            ;SEEK !
 04546 000000          0                ;CYLINDER 0
 04547 006242          EHALT            ;ERROR IN SEEK, AC1=STATUS
 04550 000411          JMP E15E         ;SKIP TO END OF TEST
 04551 006256          GENDAT           ;GENERATE DATA
 04552 005434          HDN              ;ADDRESS OF DATA GENERATOR
 04553 006605          PRGEND           ;DATA BUFFER ADDRESS
 04554 006260          WRITE            ;WRITE !
 04555 006605          PRGEND           ;DATA BUFFER ADDRESS
 04556 000017 E15.1:   17               ;DISK ADDRESS (IT CHANGES)
 04557 006242          EHALT            ;ERROR IN WRITE, AC1=STATUS
 04560 000401          JMP .+1
 04561 006243 E15E:    LOOP

 04562 040156          ISZ HEAD         ;INCREMENT TO NEXT HEAD
 04563 020773          LDA 0,E15.1
 04564 024067          LDA 1,K87
 04565 123000          ADD 1,0
 04566 040770          STA 0,E15.1
 04567 024150          LDA 1,DTYPE      ;DONE ?
 04570 030454          LDA 2,ADR-1
 04571 125112          MOVL# 1,1,SZC
 04572 000404          JMP .+4          ;CART DISK
 04573 151400          INC 2,2
 04574 125202          MOVR 1,1,SZC     ;SKIP IF 2311
 04575 151400          INC 2,2          ;2314
 04576 025000          LDA 1,0,2        ;GET END DISK ADDR
 04577 122404          SUB 1,0,SZR
 04600 000744          JMP E15          ;NOT DONE YET
```

```
▲ 0662  .MAIN

04601 102400          SUB 0,0           ;SEE THE DISCUSSION PRECEDING
04602 040156          STA 0,HEAD        ;TEST E15.  READ ONCE FROM
04603 020121          LDA 0,C17         ;EACH HEAD ON CYL-0, SECT-0.
04604 040410          STA 0,E16.2       ;VERIFY THAT DATA=HEAD #
04605 006237 E16:     JSR @ISET
04606 006262          DOSEK             ;SEEK !
04607 000000          0                 ;CYLINDER 0
04610 006242          EHALT             ;ERROR IN SEEK, AC1=STATUS
04611 000412          JMP F16E          ;SKIP TO END OF TEST
04612 006257          READ              ;READ !
04613 006605 E16.1:   PRGEND            ;ADDRESS OF DATA BUFFER
04614 000017 E16.2:   17                ;DISK ADDRESS (IT CHANGES)
04615 006242          EHALT             ;ERROR IN READ, AC1=STATUS
04616 000405          JMP F16E          ;SKIP TO END OF TEST
04617 020156          LDA 0,HEAD        ;GET HEAD #
04620 026773          LDA 1,@E16.1      ;GET A WORD READ
04621 122414          SUB# 1,0,SZR      ;DATA ERROR, SEE ABOVE DISCUSSION
04622 006242          EHALT             ;AC0=GOOD WORD
04623 006243 F16E:    LOOP              ;AC1=BAD

04624 010156          ISZ HEAD          ;INCREMENT TO NEXT HEAD
04625 020767          LDA 0,E16.2
04626 024067          LDA 1,KH7
04627 123000          ADD 1,0
04630 040764          STA 0,E16.2
04631 020150          LDA 1,DTYPE       ;DONE YET ??
04632 030412          LDA 2,ADR-1
04633 125112          MOVL# 1,1,SZC
04634 000404          JMP .+4           ;CART DISK
04635 151400          INC 2,2
04636 125202          MOVR 1,1,SZC      ;SKIP IF 2311
04637 151400          INC 2,2           ;2314
04640 025000          LDA 1,0,2         ;GET ENDING DISK ADDRESS
04641 122404          SUB 1,0,SZR
04642 000743          JMP E16           ;MORE TO GO
04643 000405          JMP .+5           ;GO TO NEXT TEST

04644 004645          .+1
04645 001017 ADR:     1017              ;CART,  HEAD 2  SECT 0  1 SECT
04646 005017          5017              ;2311,  HEAD 12 SECT 0  1 SECT
04647 012017          12017             ;2314,  HEAD 24 SECT 0  1 SECT
```

```
04650 02n150          LDA 0,DTYPE       ;SEE THE DISCUSSION
04651 03n506          LDA 2,ADR1-1      ;PRECEDING TEST E15.
04652 101112          MOVL# 0,0,SZC     ;WRITE ON CYL-0, SECT-0,
04653 000404          JMP .+4           ;ONCE FOR EACH HEAD.
04654 151400          INC 2,2           ;DATA = HEAD #
04655 101202          MOVR 0,0,SZC
04656 151400          INC 2,2
04657 025000          LDA 1,0,2         ;STARTING DISK ADDRESS
04660 044416          STA 1,E17.1       ;SELECT HEADS IN REVERSE ORDER
04661 02n213          LDA 0,C1774
04662 123700          ANDS 1,0
04663 04n156          STA 0,HEAD        ;CURRENT HEAD #
04664 006237 E17:     JSR @ISET
04665 006262          DOSEK             ;SEEK !
04666 000000          0                 ;CYLINDER 0
04667 006242          FHALT             ;ERROR IN SEEK, AC1=STATUS
04670 000411          JMP E17E          ;SKIP TO END OF TEST
04671 006256          GENDAT            ;GENERATE DATA
04672 005432          HDNC              ;ADDRESS OF DATA GENERATOR
04673 006605          PRGEND            ;DATA BUFFER ADDRESS
04674 006260          WRITE             ;WRITE !
04675 006605          PRGEND            ;DATA BUFFER ADDRESS
04676 000017 E17.1:   17                ;DISK ADDRESS (IT CHANGES)
04677 006242          FHALT             ;ERROR IN WRITE, AC1=STATUS
04700 000401          JMP .+1
04701 006243 E17E:    LOOP

04702 014156          DSZ HEAD
04703 000401          JMP .+1
04704 02n772          LDA 0,E17.1       ;DECREMENT HEAD #
04705 024067          LDA 1,KB7
04706 122400          SUB 1,0
04707 040767          STA 0,E17.1
04710 101103          MOVL 0,0,SNC      ;DONE YET ?
04711 000753          JMP E17           ;NO
```

```
04712 020150          LDA 0,DTYPE     ;SEE THE DISCUSSION PRECEDING
04713 030444          LDA 2,ADR1-1    ;TEST E15.  READ ONCE FROM
04714 101112          MOVL# 0,0,SZC   ;EACH HEAD ON CYL-0, SECT-0.
04715 000404          JMP .+4         ;VERIFY THAT DATA=HEAD #
04716 151400          INC 2,2
04717 101202          MOVR 0,0,SZC    ;SKIP IF 2311
04720 151400          INC 2,2         ;2314
04721 025000          LDA 1,0,2       ;GET DISK ADDRESS TO BEGIN WITH
04722 044413          STA 1,E18.2
04723 020213          LDA 0,C1774
04724 123700          ANDS 1,0
04725 040156          STA 0,HEAD      ;BEGINNING HEAD #
04726 006237 E18:     JSR @ISET
04727 006262          DOSEK           ;SEEK !
04730 000000          0               ;CYLINDER 0
04731 006242          EHALT           ;ERROR IN SEEK, AC1=STATUS
04732 000413          JMP E18E        ;SKIP TO END OF TEST
04733 006257          READ            ;READ !
04734 006605 E18.1:   PRGEND          ;ADDRESS OF DATA BUFFER
04735 000017 E18.2:   17              ;DISK ADDRESS (IT CHANGES)
04736 006242          EHALT           ;ERROR IN READ, AC1=STATUS
04737 000406          JMP E18E        ;SKIP TO END OF TEST
04740 020156          LDA 0,HEAD      ;GET HEAD #
04741 100000          COM 0,0         ;USE THE COMP.
04742 026772          LDA 1,@E18.1    ;GET A WORD READ
04743 122414          SUB# 1,0,SZR    ;DATA ERROR, SEE ABOVE DISCUSSION
04744 006242          EHALT           ;AC0=GOOD WORD
04745 006243 E18E:    LOOP            ;AC1=BAD

04746 014156          DSZ HEAD
04747 000401          JMP .+1
04750 020765          LDA 0,E18.2     ;DECREMENT HEAD #
04751 024067          LDA 1,KB7
04752 122400          SUB 1,0
04753 040762          STA 0,E18.2
04754 101103          MOVL 0,0,SNC    ;DONE YET ?
04755 000751          JMP E18         ;NO
04756 000405          JMP E19         ;YES, GO TO NEXT TEST

04757 004760          .+1
04760 000417 ADR1:    417             ;CART,  HEAD 1  SECT 0  1 SECT
04761 004417          4417            ;2311,  HEAD 11 SECT 0  1 SECT
04762 011417          11417           ;2314,  HEAD 23 SECT 0  1 SECT
```

```
04763 006240 E19:    SETP1            ;CAUSE SEEK ERROR BY
04764 006262         DOSEK            ;SEEKING TO CYL 313
04765 000313         313
04766 000403         JMP .+3
04767 000402         JMP .+2          ;AC1=STATUS
04770 006242         EHALT            ;NO ERRORS ON SEEK
04771 006243         LOOP             ;TO CYL 313

04772 030153         LDA 2,UNUM       ;RECALIBRATE THE
04773 021113         LDA 0,TRCL,2     ;UNIT WITH A SEEK ERR
04774 040401         STA 0,.+1
04775 006231         RECL0

04776 006240 E20:    SETP1            ;CAUSE SEEK ERROR BY
04777 006262         DOSEK            ;SEEKING TO CYL 313
05000 000313         313
05001 020064         LDA 0,K810
05002 123415         AND# 1,0,SNR     ;AC1=STATUS
05003 006242         EHALT            ;NO SEEK ERROR STATUS
05004 006243         LOOP             ;FOLLOWING SEEK TO CYL 313

05005 030153         LDA 2,UNUM       ;RECALIBRATE THE UNIT
05006 021113         LDA 0,TRCL,2     ;WITH A SEEK ERROR
05007 040401         STA 0,.+1
05010 006231         RECL0

05011 006240 E21:    SETP1            ;CHECK FOR ILLEGAL ERRORS
05012 006262         DOSEK            ;ALONG WITH SEEK ERROR
05013 000313         313              ;AC1=STATUS
05014 020177         LDA 0,C36        ;INTENTIONAL SEEK ERROR
05015 123414         AND# 1,0,SZR     ;ILLEGAL STATUS; "END CYL"
05016 006242         EHALT            ;OR "UNSAFE" OR "CHK WD"
05017 006243         LOOP             ;OR "DATA LATE"

05020 030153         LDA 2,UNUM       ;RECALIBRATE THE UNIT
05021 021113         LDA 0,TRCL,2     ;WITH A SEEK ERROR
05022 040401         STA 0,.+1
05023 006231         RECL0
```

```
                                   5722
05024 006254 E22:       DORW  5072         ;SETUP SECTOR 3 & 4
05025 000000            0                  ;CYL #
05026 005405            THREE              ;DATA TYPE
05027 000077            77                 ;DISK ADDRESS

05030 006254 5722       DORW               ;SECT 4
05031 000000            0
05032 005407            FOUR
05033 000117            117

05034 006237            JSR @ISET          ;TEST READ SECT 3-WRITE SECT 4
                                           ;SEQUENCE
05035 006257 5276       READ               ;READ A SECTOR
05036 006605            PRGEND             ;MEM BUFFER ADDRESS
05037 000077            77                 ;SECTOR THREE
05040 006242            EHALT              ;ERROR, AC1=STATUS
05041 000422            JMP E22E           ;SKIP TO END OF TEST

05042 006260            WRITE              ;WRITE A SECTOR
05043 006605            PRGEND             ;BUFF ADDR (3'S JUST READ)
05044 000117            117                ;SECTOR 4
05045 006242            EHALT              ;ERROR, AC1=STATUS
05046 000415            JMP E22E           ;SKIP TO END OF TEST

05047 006256            GENDAT             ;GENERATE DATA BUFFER
05050 005405            THREE              ;ADDR OF DATA GEN
05051 006605            PRGEND             ;BUFFER ADDR.

05052 006257            READ               ;READ A SECTOR
05053 007205            PRGEND+400         ;BUFF ADDR.
05054 000117            117                ;SECTOR 4
05055 006242            EHALT              ;ERROR, AC1=STATUS
05056 000405            JMP E22E           ;SKIP TO END OF TEST

05057 006261            CHECK              ;COMPARE BUFFERS A/B
05060 006605            PRGEND             ;ADDR OF BUFF A (CORRECT)
05061 007205            PRGEND+400         ;ADDR OF BUFF B
05062 006242            EHALT              ;COMPARE ERROR, AC0=GOOD
05063 006243 E22E:      LOOP               ;AC1=BAD
```

```
05064 006254 E23:   DORW            ;SETUP SECTOR 3 & 4
05065 000000        0               ;CYL # 0
05066 005405        THREE           ;DATA TYPE
05067 000077        77              ;DISK ADDRESS (SECT 3)

05070 006254        DORW
05071 000000        0
05072 005407        FOUR
05073 000117        117             ;SECTOR 4

05074 006237        JSR @ISET       ;TEST WRITE SECT 3-READ SECT 4
                                    ;SEQUENCE.
05075 006256        GENDAT          ;GENERATE DATA
05076 005407        FOUR            ;4'S
05077 006605        PRGEND          ;ADDR OF DATA BUFF

05100 006260        WRITE           ;WRITE A SECTOR
05101 006605        PRGEND          ;ADDR OF DATA BUFF
05102 000077        77              ;SECTOR 3
05103 006242        EHALT           ;ERROR, AC1=STATUS
05104 000417        JMP E23E        ;SKIP TO END OF TEST

05105 006257        READ            ;READ A SECTOR
05106 007205        PRGEND+400      ;MEM BUFF ADDRESS
05107 000117        117             ;SECTOR 4
05110 006242        EHALT           ;ERROR, AC1=STATUS
05111 000412        JMP E23E        ;SKIP TO END OF TEST

05112 006257        READ            ;READ A SECTOR
05113 006605        PRGEND          ;MEM ADDR
05114 000077        77              ;SECTOR 3
05115 006242        EHALT           ;ERROR, AC1=STATUS
05116 000405        JMP E23E        ;SKIP TO END OF TEST

05117 006261        CHECK           ;COMPARE SECTOR 3 & 4
05120 006605        PRGEND          ;SECT 3
05121 007205        PRGEND+400      ;SECT 4
05122 006242        EHALT           ;COMPARE ERROR, AC0=GOOD (SECT 3)
05123 006243 E23E:  LOOP            ;AC1=BAD (SECT 4)
```

```
▲ 0068  .MAIN

05124 102400 E24:    SUB 0,0          ;WRITE CYL # INTO EACH
05125 040405         STA 0,.CL        ;SECT 0, HEAD 0, OF ALL
05126 040420         STA 0,.SCYL      ;CYLINDERS
05127 020007         LDA 0,C312
05130 040434         STA 0,LCYL
05131 006254         DORW
05132 000000 .CL:    0                ;CYL #
05133 005417         CYLN             ;ADDR OF DATA GEN ROUT.
05134 000017         17               ;DISK ADDRESS

05135 010775         ISZ .CL
05136 020774         LDA 0,.CL
05137 024207         LDA 1,C312
05140 122437         SUBZ# 1,0,SBN
05141 000770         JMP .CL-1
05142 020454         LDA 0,C500
05143 040452         STA 0,CCNT       ; DO 500 SEEKS

05144 006237 GO:     JSR @ISET
05145 006262         DOSEK            ;SEEK !!
05146 000000 .SCYL:  0                ;CYL #
05147 006242         EHALT            ;ERROR, AC1=STATUS
05150 000412         JMP E24E         ;SKIP TO END OF TEST

05151 006257         READ             ;READ DATA IN SECT 0
05152 006605         PRGEND           ;MEM ADDR
05153 000017         17               ;DISK ADDR
05154 006242         EHALT            ;ERROR, AC1=STATUS
05155 000405         JMP E24E         ;SKIP TO END OF TEST

05156 020155         LDA 0,CYL        ;CURRENT CYLINDER #
05157 026160         LDA 1,@BUFF      ;ACTUAL CYL #
05160 122414         SUB# 1,0,SZR     ;(LCYL) = LAST CORRECT
05161 006242         EHALT            ;CYLINDER #
05162 006243 E24E:   LOOP
05163 101001         MOV 0,0,SKP
05164 000000 LCYL:   0
05165 020761         LDA 0,.SCYL      ;SAVE CYLINDER JUST DONE
05166 040776         STA 0,LCYL
05167 006255 GRAN:   JSR @IRAN        ;GET RANDOM CYLINDER #
05170 024207         LDA 1,C312
05171 030125         LDA 2,C377       ;# MUST BE <313
05172 143400         AND 2,0
05173 106433         SUBZ# 0,1,SNC
05174 000773         JMP GRAN         ;TRY AGAIN
05175 040751         STA 0,.SCYL
05176 014417         DSZ CCNT
05177 000745         JMP GO
```

```
▲ 0069  .MAIN

 05200 006247          PCRLF             ;END TEST
 05201 006250          MESSAGE
 05202 006602          MSG6              ;"PASS"
 05203 030045          LDA 2,45
 05204 025000          LDA 1,0,2
 05205 125005          MOV 1,1,SNR
 05206 002406          JMP 0,+6
 05207 015003          DSZ 3,2
 05210 002404          JMP 0,+4
 05211 060277          INTDS
 05212 035004          LDA 3,4,2
 05213 001400          JMP 0,3
 05214 000412          A1

 05215 000000 CCNT:    0
 05216 000764 C500:    500.

                       .EOT
```

```
0070  .MAIN

                        ;RANDOM NUMBER GENERATOR

05217 054431 RAN:       STA 3,.UD03     ;GENERATE A RANDOM
05220 050427            STA 2,.UD02
05221 044425            STA 1,.UD01
05222 020142            LDA 0,RANDOM    ;NUMBER IN AC0
05223 004410            JSR .UD50
05224 034426            LDA 3,.UD20
05225 163000            ADD 3,0
05226 040142            STA 0,RANDOM    ;STORE NEW VALUE.
05227 111100            MOVL 0,2
05230 030417            LDA 2,.UD02
05231 024415            LDA 1,.UD01
05232 002416            JMP @.UD03

05233 024420 .UD50:     LDA 1,.UD21     ;RANDOM CONTINUED
05234 044415            STA 1,.UD10
05235 105120            MOVZL 0,1
05236 125120            MOVZL 1,1
05237 014412            DSZ .UD10
05240 000776            JMP .-2
05241 107000            ADD 0,1
05242 125120            MOVZL 1,1
05243 125120            MOVZL 1,1
05244 123000            ADD 1,0
05245 001400            JMP 0,3
05246 000000 .UD01:     0
05247 000000 .UD02:     0
05250 000000 .UD03:     0
05251 000000 .UD10:     0
05252 033031 .UD20:     33031
05253 000010 .UD21:     10
```

```
                        ;CHECK DATA SUBROUTINE
                        ; CALL  CHECK
                        ;       ADDRESS OF DATA BUFFER 1
                        ;       ADDRESS OF DATA BUFFER 2
                        ;       ERROR RETURN, (AC1)=BAD (AC0)=GOOD
                        ;       NORMAL RETURN

05254 054460 .CHECK:  STA 3,GENRET
05255 030214          LDA 2,M400
05256 050417          STA 2,CTR
05257 031400          LDA 2,0,3
05260 035401          LDA 3,1,3
05261 010453          ISZ GENRET
05262 010452          ISZ GENRET
05263 021000 .CHE1:   LDA 0,0,2
05264 025400          LDA 1,0,3
05265 106414          SUB# 0,1,SZR
05266 002446          JMP @GENRET     ;ERROR
05267 151400          INC 2,2
05270 175400          INC 3,3
05271 010404          ISZ CTR
05272 000771          JMP .CHE1       ;CHECK MORE
05273 010441          ISZ GENRET
05274 002440          JMP @GENRET     ;NORMAL RETURN
05275 000000 CTR:     0

                        ;GENERATE ONE SECTOR OF DATA
                        ; CALL  GENDAT
                        ;       ADDRESS OF DATA GEN ROUTINE
                        ;       DATA BUFFER ADDRESS
                        ;       RETURN

05276 054436 .GEN:    STA 3,GENRET
05277 024214          LDA 1,M400
05300 031401          LDA 2,1,3
05301 034433 .GEN1:   LDA 3,GENRET
05302 007400          JSR @0,3        ;GET A DATA WORD
05303 041000          STA 0,0,2
05304 151400          INC 2,2
05305 125404          INC 1,1,SZR
05306 000773          JMP .GEN1       ;DO MORE
05307 034425          LDA 3,GENRET    ;DONE
05310 001402          JMP 2,3
```

```
▲ 0072  .MAIN

                        ;WRITE SUBROUTINE
                        ; CALL  WRITE
                        ;       DATA BUFFER ADDRESS
                        ;       DISK ADDRESS
                        ;       ERROR RETURN, (AC1) = STATUS
                        ;       JMP TO END OF TEST
                        ;       NORMAL RETURN

                        ;ERROR RETURN IF
                        ;       TIMEOUT (100MS)
                        ;       DATA LATE
                        ;       ADDRESS ERROR/UNSAFE
                        ;       END CYLINDER
                        ;       SEEK ERROR
                        ;       ANY "SEEKING"
                        ;       ANY "SEEK DONE"
                        ;       NO "R/W DONE"

005311 054423 .WRITE: STA 3,GENRET
005312 021400         LDA 0,0,3
005313 062033         DOB 0,DSKP        ;CA
005314 020152         LDA 0,TESTU
005315 025401         LDA 1,1,3
005316 123000         ADD 1,0
005317 063233         DOCC 0,DSKP       ;UNIT # & DISK ADDRESS
005320 020126         LDA 0,C777        ;BIT 7 AND ALL CYL BITS=1
005321 061133         DOAS 0,DSKP       ;WRITE !
005322 010412         ISZ GENRET
005323 010411         ISZ GENRET
005324 006235         ITRWT             ;WAIT 100MS FOR INTERRUPT
005325 002407         JMP @GENRET       ;TIMEOUT
005326 020407 .WR1:   LDA 0,.WM         ;ERROR BIT MASK
005327 107415         AND# 0,1,SNR
005330 125113         MOVL# 1,1,SNC
005331 002403         JMP @GENRET       ;ERROR STATUS
005332 034402         LDA 3,GENRET
005333 001402         JMP 2,3

005334 000000 GENRET: 0
005335 077677 .WM:    77677
```

```
                        ;READ SUBROUTINE
                        ; CALL  READ
                        ;       DATA BUFFER ADDRESS
                        ;       DISK ADDRESS
                        ;       ERROR RETURN, (AC1)=STATUS
                        ;       JMP TO END TEST
                        ;       NORMAL RETURN

                        ;ERROR RETURN IF
                        ;       TIMEOUT (100MS)
                        ;       DATA LATE
                        ;       CHECK WORD ERROR
                        ;       ADDRESS ERROR/UNSAFE
                        ;       END CYLINDER
                        ;       SEEK ERROR
                        ;       ANY "SEEKING"
                        ;       ANY "SEEK DONE"
                        ;       NO "R/W DONE"

05336 054776 .READ:     STA 3,GENRET
05337 021400            LDA 0,0,3
05340 062033            DOB 0,DSKP          ;CA
05341 020152            LDA 0,TESTU
05342 025401            LDA 1,1,3
05343 123000            ADD 1,0
05344 063233            DOCC 0,DSKP         ;UNIT # & DISK ADDRESS
05345 102400            SUB 0,0
05346 061133            DOAS 0,DSKP         ;READ !
05347 010765            ISZ GENRET
05350 010764            ISZ GENRET
05351 006235            ITRWT               ;WAIT 100 MS FOR INTERRUPT
05352 002762            JMP @GENRET         ;TIMEOUT
05353 000753            JMP .WR1            ;CHECK STATUS
```

```
                        ;SEEK SUBROUTINE
                        ; CALL  DOSEK
                        ;       CYL #
                        ;       ERROR RETURN (AC1)=STATUS
                        ;       JMP TO END TEST
                        ;       NORMAL RETURN

                        ;ERROR RETURN IF
                        ;       TIMEOUT (500MS)
                        ;       DATA LATE
                        ;       CHECK WORD ERROR
                        ;       ADDRESS ERROR/UNSAFE
                        ;       END CYLINDER
                        ;       SEEK ERROR
                        ;       ANY "SEEKING"
                        ;       NO "SEEK DONE"
                        ;       "R/W DONE"

005354 054760 .DOSEK: STA 3,GENRET
005355 010757         ISZ GENRET
005356 020152         LDA 0,TESTU
005357 063233         DOCC 0,DSKP       ;SELECT UNIT
005360 021400         LDA 0,0,3
005361 024125         LDA 1,C377
005362 123400         AND 1,0
005363 040155         STA 0,CYL         ;SAVE CYL #
005364 024070         LDA 1,K66
005365 123000         ADD 1,0           ;CYL # + SEEK
005366 061333         DOAP 0,DSKP
005367 030167         LDA 2,C5
005370 006227         WAIT              ;WAIT 500MS (OR UNTIL "DONE")
005371 020410         LDA 0,.DM
005372 123414         AND# 1,0,SZR
005373 002741         JMP @GENRET       ;ERROR STATUS
005374 020220         LDA 0,C74K
005375 123415         AND# 1,0,SNR
005376 002736         JMP @GENRET       ;NO SEEK DONE
005377 034735         LDA 3,GENRET
005400 001402         JMP 2,3

005401 103677 .DM:    103677
```

```
▲ 0075  .MAIN

05402 102001 ONES:   ADC 0,0,SKP
05403 102400 ZEROS:  SUB 0,0
05404 001400         JMP 0,3

05405 020117 THREE:  LDA 0,C3
05406 001400         JMP 0,3

05407 020166 FOUR:   LDA 0,C4
05410 001400         JMP 0,3

05411 020140 ALT1:   LDA 0,C2525
05412 001400         JMP 0,3

05413 020141 ALT0:   LDA 0,C5252
05414 001400         JMP 0,3

05415 020155 CYLNC:  LDA 0,CYL
05416 100001         COM 0,0,SKP
05417 020155 CYLN:   LDA 0,CYL
05420 001400         JMP 0,3

05421 054407 NUMSEQ: STA 3,NSRET
05422 010407         ISZ NS1
05423 000401         JMP .+1
05424 020405         LDA 0,NS1
05425 034125         LDA 3,C377
05426 163400         AND 3,0
05427 002401         JMP @NSRET
05430 000000 NSRET:  0
05431 000000 NS1:    0

05432 020156 HDNC:   LDA 0,HEAD
05433 100001         COM 0,0,SKP
05434 020156 HDN:    LDA 0,HEAD
05435 001400         JMP 0,3

05436 020157 SETNC:  LDA 0,SECT
05437 100001         COM 0,0,SKP
05440 020157 SECTN:  LDA 0,SECT
05441 001400         JMP 0,3
```

```
                        ;PROGRAM INITILAIZATION

05442 054504 .INI:   STA 3,.INRET
05443 006247         PCRLF
05444 006250         MESSAGE
05445 006510         MSG1              ;DISK PACK TYPE
05446 006247         PCRLF
05447 006250         MESSAGE
05450 006523         MSG2              ;TYPE 0 FOR CART
05451 006247         PCRLF             ;      1 FOR 2311
05452 006250         MESSAGE           ;      2 FOR 2314
05453 006541         MSG3

05454 006247 .INI1:  PCRLF
05455 060210         NIOC TTI          ;WAIT FOR TTI INPUT
05456 063610         SKPDN TTI
05457 000777         JMP .-1
05460 060410         DIA 0,TTI         ;READ CHAR
05461 061111         DOAS 0,TTO        ;ECHO IT
05462 024124         LDA 1,C177        ;7 BIT MASK
05463 107400         AND 0,1
05464 030200         LDA 2,C60
05465 034201         LDA 3,C63
05466 132437         SUBZ# 1,2,SBN     ;CHAR MUST 0,1, OR 2
05467 166432         SUBZ# 3,1,SZC
05470 000457         JMP QUEST         ;ILLEGAL CHAR TYPED
05471 024117         LDA 1,C3
05472 123400         AND 1,0
05473 101224         MOVZR 0,0,SZR
05474 000404         JMP .+4           ;=2, 2314, BIT 15 IS SET, DONE
05475 101205         MOVR 0,0,SNR
05476 101241         MOVOR 0,0,SKP     ;=0, CART, MUST SET BIT 0
05477 101220         MOVZR 0,0         ;=1, 2311, MUST SET BIT 1
05500 040150         STA 0,DTYPE
```

```
05501 102400            SUB 0,0
05502 040151            STA 0,NDSKS
05503 006247 .INI2:     PCRLF
05504 006250            MESSAGE
05505 006562            MSG5                ;TYPE UNIT NUMBERS TO TEST
05506 006247            PCRLF

05507 060210 .INI3:     NIOC TTI
05510 063610            SKPDN TTI
05511 000777            JMP .-1
05512 060410            DIA 0,TTI           ;READ CHAR TYPED
05513 061111            DOAS 0,TTO          ;ECHO THE CHAR
05514 024124            LDA 1,C177
05515 107400            AND 0,1             ;7 BIT ASCII
05516 030173            LDA 2,C15
05517 132415            SUB# 1,2,SNR
05520 000422            JMP CR              ;CR TYPED
05521 030200            LDA 2,C60
05522 034201            LDA 3,C63
05523 166437            SUBZ# 3,1,SBN       ;# MUST BE 0,1,2, OR 3
05524 132436            SUBZ# 1,2,SEZ
05525 000426            JMP QST1            ;ILLEGAL CHAR TYPED
05526 030117            LDA 2,C3
05527 113400            AND 0,2
05530 021057            LDA 0,UNTBIT,2
05531 025053            LDA 1,.TU,2
05532 044152            STA 1,TESTU
05533 050153            STA 2,UNUM
05534 110000            COM 0,2
05535 024151            LDA 1,NDSKS
05536 147400            AND 2,1             ;(NDSKS) = BIT 15 - UNIT 0
05537 107000            ADD 0,1             ;          BIT 14 - UNIT 1
05540 044151            STA 1,NDSKS         ;          BIT 13 - UNIT 2
05541 000746            JMP .INI3           ;          BIT 12 - UNIT 3

05542 020151 CR:        LDA 0,NDSKS         ;IF NO UNIT NUMBERS TYPED
05543 101005            MOV 0,0,SNR         ;IT IS AN ERROR
05544 000407            JMP QST1
05545 002401            JMP @.INRET
05546 000000 .INRET:    0

05547 006247 QUEST:     PCRLF
05550 006250            MESSAGE
05551 006557            MSG4
05552 000732            JMP .INI1

05553 006247 QST1:      PCRLF
05554 006250            MESSAGE
05555 006557            MSG4
05556 000725            JMP .INI2
```

```
^ 0078  .MAIN

                        ;SET ALL I/O ADDRESSES

05557 102401 .S33:   SUB 0,0,SKP      ;SET TO 33
05560 020064 .S73:   LDA 0,KB10       ;SET TO 73
05561 024176         LDA 1,C33
05562 123000         ADD 1,0
05563 040154         STA 0,CDSK
05564 030425         LDA 2,FIRST
05565 021000 .SN0:   LDA 0,0,2
05566 024161         LDA 1,MSK1
05567 107400         AND 0,1
05570 034162         LDA 3,DPIO
05571 166404         SUB 3,1,SZR
05572 000406         JMP .SN1         ;GO ON
05573 024163         LDA 1,MSK2       ;DISK PACK IO INST.
05574 034154         LDA 3,CDSK
05575 123400         AND 1,0
05576 163000         ADD 3,0
05577 041000         STA 0,0,2
05600 151400 .SN1:   INC 2,2
05601 024411         LDA 1,LAST
05602 132414         SUB# 1,2,SZR
05603 000762         JMP .SN0
05604 024046         LDA 1,EGGS
05605 125005         MOV 1,1,SNR
05606 063077         HALT
05607 000401         JMP @.+1
05610 000400         START

05611 000400 FIRST:  START
05612 006027 LAST:   .SET

                        ;DELAY SUBROUTINE

05613 054006 .STL:   STA 3,.STLRET
05614 034050         LDA 3,KB14
05615 054000         STA 3,0
05616 014000         DSZ 0
05617 000777         JMP .-1
05620 000401         JMP @.STLRET
05621 000000 .STLRET:0
```

```
                        ;"SET A SEEKING FLOP" SUBROUTINE
                        ;FOR THE UNITS INDICATED BY (AC2).
                        ;       BIT 15 = UNIT 0
                        ;       BIT 14 = UNIT 1
                        ;       BIT 13 = UNIT 2
                        ;       BIT 12 = UNIT 3

005622 020056 .SSEK:    LDA 0,C140K
005623 151005           MOV 2,2,SNR
005624 001400           JMP 0,3             ;DONE
005625 024054           LDA 1,KB1
005626 123000           ADD 1,0
005627 151223           MOVZR 2,2,SNC
005630 000773           JMP .SSEK+1         ;TRY ANOTHER UNIT
005631 063033           DOC 0,DSKP          ;RECAL THIS UNIT
005632 024070           LDA 1,C1000
005633 065333           DOAP 1,DSKP         ;SEEK !!
005634 000767           JMP .SSEK+1

                        ;WAIT ROUTINE FOR "DONE"
                        ;RUN TIMER
                        ;(AC2) = MAX RUN TIME. (100MS/COUNT)

005635 050164 .WAIT:    STA 2,ITRCNT        ;# 100MS ITERATIONS
005636 030145           LDA 2,TIME          ;100 MS / COUNT
005637 050144           STA 2,TEMP          ;TEMP COUNTER
005640 030221           LDA 2,C174K         ;DISK DONE FLAG MASK
005641 063700           SKPDZ 0             ;DUMMY FOR TIME FILLER
005642 063077           HALT
005643 064433           DIA 1,DSKP          ;READ STATUS
005644 133414           AND# 1,2,SZR
005645 000406           JMP WTO             ;SOMEONE IS DONE
005646 014144           DSZ TEMP
005647 000772           JMP .-6
005650 014164           DSZ ITRCNT          ;DONE 100 MS
005651 000765           JMP .WAIT+1         ;DO SOME MORE
005652 001400           JMP 0,3             ;TIME OUT
005653 064433 WTO:      DIA 1,DSKP
005654 001400           JMP 0,3
```

```
▲ 0060  .MAIN

                        ;FIND COMPUTER TIME BASE

05655 062677 STB:       IORST
05656 126400            SUB 1,1
05657 044145            STA 1,TIME          ;TIME BASE 1
05660 044146            STA 1,TIME1         ;TIME BASE 1'
05661 020125            LDA 0,C377
05662 061111            DOAS 0,TTO          ;TYPE ONCE TO SYNC TTO
05663 063611            SKPDN TTO
05664 000777            JMP .-1
05665 061111            DOAS 0,TTO          ;START TIME BASE 1
05666 063711            SKPDZ TTO
05667 000467            JMP STB1            ;TIME BASE 1 ESTABLISHED
05670 064400            DIA 1,0             ;TIME FILLER
05671 127411            AND# 1,1,SKP        ;BFD
05672 063077            HALT                ;REAL TROUBLE
05673 010145            ISZ TIME            ;COUNT 1
05674 000772            JMP .-6
05675 063077            HALT                ;TTO CLOCK OFF!!

05676 061111 STB1:      DOAS 0,TTO          ;START TIME BASE 1'
05677 063711            SKPDZ TTO
05700 001400            JMP 0,3             ;TIME BASE 1' ESTABLISHED
05701 010146            ISZ TIME1
05702 000775            JMP .-3
05703 063077            HALT                ;TTO CLOCK OFF!!

                        ;RECALIBRATE SUBROUTINE

05704 102400 .RCL0:     SUB 0,0             ;UNIT 0 ENTRY
05705 000405            JMP .RCL3+1
05706 020054 .RCL1:     LDA 0,KB1           ;UNIT 1 ENTRY
05707 000403            JMP .RCL3+1
05710 102621 .RCL2:     SUBZR 0,0,SKP       ;UNIT 2 ENTRY
05711 020056 .RCL3:     LDA 0,C140K         ;UNIT 3 ENTRY
05712 063233            DOCC 0,DSKP         ;SELECT UNIT
05713 020215            LDA 0,C1400
05714 061333            DOAP 0,DSKP         ;RECALIBRATE
05715 054404            STA 3,RCLRET
05716 030173            LDA 2,C15
05717 006227            WAIT                ;WAIT 1.3SEC (OR UNTIL DONE)
05720 002401            JMP @RCLRET
05721 000000 RCLRET:    0
```

```
                        ;SEEK WRITE/READ/CHECK SUBROUTINE
                        ;CALL   DORW
                        ;.      CYL#
                        ;       ADDRESS OF DATA GENERATOR
                        ;       DISK ADDRESS
                        ;       RETURN

05722 054441 .DORW:  STA 3,DRWRET
05723 021400         LDA 0,0,3       ;XFER CYL #
05724 040410         STA 0,.D0
05725 021401         LDA 0,1,3       ;XFER ADDR DAT GEN
05726 040412         STA 0,.D01
05727 021402         LDA 0,2,3       ;XFER DISK ADDR
05730 040414         STA 0,.D02
05731 040420         STA 0,.D03

05732 006237         JSR @ISET
05733 006262         DOSEK           ;SEEK !!
05734 000000 .D0:    0               ;CYL #
05735 006242         EHALT           ;ERROR, AC1=STATUS
05736 000422         JMP E.D0        ;SKIP TO END OF TEST

05737 006256         GENDAT          ;GENERATE DATA
05740 005402 .D01:   ONES            ;ADDRESS OF DATA GEN
05741 006605         PRGEND          ;MEM ADDR

05742 006260         WRITE           ;WRITE !!
05743 006605         PRGEND          ;MEM ADDR
05744 000017 .D02:   17              ;DISK ADDRESS
05745 006242         EHALT           ;ERROR, AC1=STATUS
05746 000412         JMP E.D0        ;SKIP TO END OF TEST

05747 006257         READ            ;READ !!
05750 007205         PRGEND+400      ;MEM ADDR
05751 000017 .D03:   17              ;DISK ADDRESS
05752 006242         EHALT           ;ERROR, AC1=STATUS
05753 000405         JMP E.D0        ;SKIP TO END OF TEST

05754 006261         CHECK           ;COMPARE BUFFER A/B
05755 006605         PRGEND          ; A
05756 007205         PRGEND+400      ; B
05757 006242         EHALT           ;COMPARE ERROR, AC0=GOOD
05760 006243 E.D0:   LOOP            ;AC1=BAD

05761 034402         LDA 3,DRWRET
05762 001403         JMP 3,3

05763 000000 DRWRET: 0
```

```
                        ;WAIT FOR INTERRUPT TIMEOUT AFTER 100 MS
                        ;RETURN+1  IF TIMEOUT
                        ;RETURN+2  IF INTERRUPT

05764 020146 .IWT:      LDA 0,TIME1
05765 040144            STA 0,TEMP
05766 060177            INTEN
05767 063700            SKPDZ 0             ;TIME FILLER
05770 063077            HALT
05771 014144            DSZ TEMP
05772 000775            JMP .-3
05773 060277            INTDS               ;NO INTERRUPT
05774 064433            DIA 1,DSKP
05775 001400            JMP 0,3             ;ERROR RETURN

05776 064433 IRET:      DIA 1,DSKP
05777 001401            JMP 1,3             ;NORM INTERRUPT RETURN

                        ;CHOOSE AN ACTIVE DISK UNIT
                        ;RETURN WITH UNIT # IN AC2
                        ;       UNIT 0 = 1
                        ;       UNIT 1 = 2
                        ;       UNIT 2 = 4
                        ;       UNIT 3 = 10

06000 152521 .ADSK:     SUBZL 2,2,SKP
06001 151120            MOVZL 2,2
06002 020151            LDA 0,NDSKS
06003 143405            AND 2,0,SNR
06004 000775            JMP .-3
06005 001400            JMP 0,3

                        ;SEEK SUBROUTINE
                        ; CALL  SEEK                ;AC2 = UNIT
                        ;       N                   ;CYL #
                        ;       RETURN              ;AC1 = STATUS

06006 054420 .SK:       STA 3,SKRET
06007 126400            SUB 1,1
06010 020054            LDA 0,K81
06011 151222            MOVZR 2,2,SZC
06012 000403            JMP .SK1
06013 107000            ADD 0,1
06014 000775            JMP .-3

06015 067033 .SK1:      DOC 1,DSKP          ;SELECT UNIT
06016 021400            LDA 0,0,3
06017 024070            LDA 1,K86
06020 123000            ADD 1,0             ;SEEK + CYL #
06021 061333            DOAP 0,DSKP
06022 030167            LDA 2,C5
06023 006227            WAIT                ;WAIT 500MS (OR UNTIL DONE)
06024 010402            ISZ SKRET           ;AC1=STATUS
06025 002401            JMP @SKRET
06026 000000 SKRET:     0
```

```
▲ 0083  .MAIN

06027 054426 .SET:   STA 3,LOOPR      ;ITERATE ONCE
06030 176520         SUBZL 3,3
06031 000406         JMP .SETUP+2

06032 054423 .SETP1: STA 3,LOOPR      ;ITERATE 5 TIMES
06033 034167         LDA 3,C5
06034 000403         JMP .SETUP+2

06035 054420 .SETUP: STA     3,LOOPR  ;ADDRESS OF TOP OF LOOP
06036 034407         LDA 3,ITR        ;THIS ROUTINE INITIALIZES
06037 054407         STA 3,ITRCT      ;EACH TEST
06040 176400         SUB 3,3
06041 054406         STA 3,ESWIT
06042 054406         STA 3,ERRCT
06043 062677         IORST            ;,I/O RESET
06044 002411         JMP @LOOPR

06045 000144 ITR:    144
06046 000000 ITRCT:  0
06047 000000 ESWIT:  0
06050 000000 ERRCT:  0
06051 000000 .RTRN:  0
06052 000000 SAV2:   0
06053 000000 SAV1:   0
06054 000000 SAV0:   0
06055 000000 LOOPR:  0
06056 000000 DSWT:   0
```

```
▲ 0004  .MAIN

 06057 054772 .LUPD:  STA 3,.RTRN      ;INTRODUCE A SHORT
 06060 034122         LDA 3,C37        ;DELAY IN THE LOOP
 06061 054000         STA 3,0          ;ROUTINE TO ALLOW THE
 06062 014000         DSZ 0            ;DISK CART. ATTENTION
 06063 000777         JMP .-1          ;LINE TO COME UP
 06064 000402         JMP .+2

 06065 054764 .LOOP:  STA 3,.RTRN      ;END OF TEST ITERATION
 06066 050764         STA 2,SAV2
 06067 044764         STA 1,SAV1
 06070 040764         STA 0,SAV0
 06071 014755         DSZ ITRCT
 06072 000430         JMP CYCTS        ;NOT 100 TIMES ITERATED
 06073 034752         LDA 3,ITR        ;RESET ITERATION CNTR
 06074 054752         STA 3,ITRCT
 06075 034752         LDA     3,ESWIT
 06076 175005         MOV     3,3,SNR
 06077 002752         JMP     @.RTRN
 06100 074477         READS 3
 06101 175120         MOVZL 3,3
 06102 175100         MOVL 3,3
 06103 175103         MOVL 3,3,SNC
 06104 000414         JMP PCENT+1
 06105 006247         PCRLF            ;PRINT CARRIAGE
 06106 024742         LDA 1,ERRCT
 06107 030736         LDA 2,ITR
 06110 004546         JSR MULT
 06111 030734         LDA 2,ITR
 06112 004530         JSR DIVD
 06113 006506         JSR @IPDEC       ;PRINT VALUE
 06114 020403         LDA 0,PCENT      ;EXAMPLE:  89%
 06115 006503         JSR @ICHAR
 06116 000402         JMP PCENT+1
```

```
06117 000045 PCENT:  "%                  ;CHARACTOR
06120 176400         SUB      3,3
06121 054727         STA      3,ERRCT
06122 034725 CYCTS:  LDA 3,ESWIT
06123 175004         MOV 3,3,SZR
06124 000410         JMP CNS             ;THERE HAS BEEN AN ERROR
06125 062677 CYC1:   IORST               ;I/O RESET
06126 020726         LDA 0,SAV0          ;RESTORE AC'S
06127 024724         LDA 1,SAV1
06130 030722         LDA 2,SAV2
06131 175113         MOVL# 3,3,SNC       ;SWITCH 0
06132 000723         JMP @LOOPR          ;(0)=LOOP ROUTINE
06133 000716         JMP @.RTRN          ;(1)=PROCEED TO NEXT TEST

06134 074477 CNS:    READS 3             ;ERROR LOOP. IS A RECAL
06135 024073         LDA 1,KB3           ;REQUESTED ?
06136 137415         AND# 1,3,SNR
06137 000410         JMP CNS1            ;NO  GO-ON
06140 175300         MOVS 3,3            ;YES, ASSEMBLE THE
06141 024170         LDA 1,C6            ;UNIT # FROM SW5-6
06142 137620         ANDZR 1,3
06143 021513         LDA 0,TRCL,3
06144 040401         STA 0,.+1           ;PICK A RECAL
06145 006234         RECL3               ;RECALIBRATE !!
06146 004524         JSR .DLY            ;DELAY 1 SEC

06147 074477 CNS1:   READS 3             ;IF SW4=1 DELAY
06150 020072         LDA 0,KB4           ; 1 SEC.
06151 117404         AND 0,3,SZR
06152 004520         JSR .DLY
06153 034703         LDA 3,DSWT          ;FORCED 2 SEC DELAY ??
06154 175005         MOV 3,3,SNR
06155 000403         JMP .+3             ;NO
06156 004514         JSR .DLY            ;YES,    1 SEC
06157 004513         JSR .DLY            ;        1 SEC
06160 074477         READS 3
06161 000744         JMP CYC1
```

```
06162 054667 .EH1:   STA 3,.RTRN      ;ERR WITH FORCED 1 SEC DELAY
06163 176520         SUBZL 3,3
06164 000403         JMP .EH2
06165 054664 .EHALT: STA 3,.RTRN      ;ERROR SUBROUTINE
06166 176400         SUB 3,3          ;ERROR WITH NO DELAY FORCED
06167 054667 .EH2:   STA 3,DSWT       ;DELAY SWITCH
06170 034657         LDA 3,ESWIT
06171 175004         MOV 3,3,SZR
06172 000423         JMP ERET
06173 034656 ERR1:   LDA 3,.RTRN

                     ;ERROR. C(3)=PC
06174 004430         JSR AUTOER       ;OPERATOR,SET SWITCHS!
06175 054652         STA 3,ESWIT

06176 074477         READS 3
06177 177112         ADDL# 3,3,SZC    ;LOOK AT SWITCH 1
06200 000415         JMP ERET

06201 040653         STA      0,SAV0
06202 044651         STA      1,SAV1
06203 006247         PCRLF            ;PRINT CARRIAGE
06204 006250         MESSAGE          ;AND HEADER
06205 006222         HEADER
06206 020643         LDA 0,.RTRN
06207 040640         STA      0,ESWIT
06210 126000         ADC 1,1
06211 107000         ADD 0,1
06212 004522         JSR PDCT         ;PC OF ERROR
06213 020641         LDA      0,SAV0
06214 024637         LDA      1,SAV1
06215 010633 ERET:   ISZ      ERRCT
06216 002633         JMP      @.RTRN
06217 002632         JMP      @.RTRN

06220 006415 ICHAR:  CHAR.
06221 006337 IPDEC:  PDEC

             HEADER: .TXTE !
06222 141520 PC      !
06223 000011


06224 024046 AUTOER: LDA 1,EGGS
06225 125004         MOV 1,1,SZR
06226 000405         JMP .+5
06227 171000         MOV 3,2
06230 034621         LDA 3,.RTRN
06231 063077         HALT
06232 001000         JMP 0,2
06233 060277         INTDS
06234 034052         LDA 3,EGGS+4
06235 001400         JMP 0,3
```

```
                ;AC1 REM AC0=(AC0,AC1)/AC2
06236 102400 DIVIN: SUB     0,0
06237 054431 DIVDO: STA     3,MSAV
06240 142432        SUBZ#   2,0,SZC
06241 000413        JMP     DEXT
06242 054426 DIVD:  STA 3,MSAV      ;DIVIDE
06243 034426        LDA 3,M20
06244 125120        MOVZL 1,1
06245 101100 DLOOP: MOVL 0,0
06246 142412        SUB# 2,0,SZC
06247 142400        SUB 2,0
06250 125100        MOVL 1,1
06251 175404        INC 3,3,SZR
06252 000773        JMP DLOOP
06253 176441        SUBO    3,3,SKP
06254 176420 DEXT:  SUBZ    3,3
06255 002413        JMP @MSAV

                ;(AC0,AC1)=AC1*AC2+AC0
06256 102460 MULT:  SUBC 0,0        ;MULTIPLY
06257 054411        STA 3,MSAV
06260 034411        LDA 3,M20
06261 125203 MLOOP: MOVR 1,1,SNC
06262 101201        MOVR 0,0,SKP
06263 143220        ADDZR 2,0
06264 175404        INC 3,3,SZR
06265 000774        JMP MLOOP
06266 125260        MOVCR 1,1
06267 002401        JMP @MSAV
06270 000000 MSAV:  0

06271 177760 M20:   -20

06272 020172 .DLY:  LDA 0,C12       ;DELAY 1 SEC
06273 040164        STA 0,ITRCNT    ;10.X100MS
06274 020145 .DLY1: LDA 0,TIME
06275 040144        STA 0,TEMP
06276 063700        SKPDZ 0         ;
06277 063077        HALT            ;
06300 060433        DIA 0,DSKP      ;
06301 103411        AND# 0,0,SKP    ; 100 MS
06302 063077        HALT            ; DELAY LOOP
06303 014144        DSZ TEMP        ;
06304 000772        JMP .-6         ;
06305 014164        DSZ ITRCNT
06306 000766        JMP .DLY1
06307 001400        JMP 0,3
```

```
                ;TTO NON INTERRUPT PACKAGE
                ;"MESS" PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLR
                ;"CHAR" PRINTS ASCII CHARACTOR, C(0)R,C(0)L MUST BE 0
                ;WILL RETURN +2 IF C(0)R=0,CORRECTS THE PARITY,33 SIMULATE
                ;"TYPE" PRINTS C(0)R. MUST HAVE PROPER PARITY. RETURN IS
                ;TO CALL+1.REPLACE THIS ROUTINE WITH INTERRUPT TYPE IF DESIREO.
                ;"CRLF" PRINTS A CARRIAGE RETURN
                ;"POCT" PRINTS C(1) IN OCTAL FOLLOWED BY A TAB
                ;"PDEC" PRINTS C(1) IN DECIMAL,LEADING ZEROS SUPPRESSED,
                ;FOLLOWED BY A TAB.
06310 054570 MESS:   STA 3,MESSR      ;PRINT A TEXT MESSAGE
06311 070477         READS 2
06312 153102         ADDL 2,2,SZC     ;NO PRINT IF SW1=1
06313 001401         JMP 1,3
06314 010564         ISZ MESSR
06315 031400         LDA 2,0,3        ;C(2) POINTS TO MESSAGE
06316 024125         LDA 1,C377       ;A 8 BIT MASK
06317 021000 MES.1:  LDA 0,0,2        ;C(2)=DATA WORD
06320 125112         MOVL# 1,1,SZC
06321 123701         ANDS 1,0,SKP
06322 123401         AND 1,0,SKP      ;C(0)=DATA CHARACTOR RIGHT
06323 151400         INC 2,2          ;INC TO NEXT WORD
06324 124000         COM 1,1          ;FLIP MASK
06325 004470         JSR CHAR.        ;PRINT
06326 000771         JMP MES.1        ;ANOTHER
06327 063511         SKPBZ TTO
06330 000777         JMP .-1
06331 060211         NIOC TTO
06332 002546         JMP @MESSR       ;LAST

06333 102401 ZOCT:   SUB 0,0,SKP
06334 020200 POCT:   LDA 0,C60
06335 030437         LDA 2,OCTAB      ;PRINT C(1) IN OCTAL
06336 000403         JMP .+3
06337 030445 PDEC:   LDA 2,DECTB      ;PRINT C(1) IN DECIMAL
06340 102400         SUB 0,0
06341 054453         STA 3,RADRET     ;BOTH ENTRYS PRINT NUMBER
06342 074477         READS 3
06343 177102         ADDL 3,3,SZC     ;NO PRINT IF SW1=1
06344 002450         JMP @RADRET
06345 040446         STA 0,ZSUPP      ;THEN TAB TO NEXT POSITION
06346 050401         STA 2,.+1
06347 000000 DECOCT: 0                ;A"LDA 2,TABLE" INSTRUCTION
06350 010777         ISZ .-1
06351 034443         LDA 3,RADRET     ;SETUP "TAB" AT END
06352 020516         LDA 0,CHTAB
06353 151005         MOV 2,2,SNR      ;IF TABLE ENTRY=0
06354 000441         JMP CHAR.        ;EXIT WITH TAB
06355 034436         LDA 3,ZSUPP      ;ZEROS SUPPRESS STUF
06356 102400         SUB 0,0
06357 146512 DECOT:  SUBL# 2,1,SZC
06360 000405         JMP DECP
06361 146400         SUB 2,1          ;FORM THE DIGIT
06362 034200         LDA 3,C60
06363 101400         INC 0,0
06364 000773         JMP DECOT
```

```
06365 151235 DECP:   MOVZR# 2,2,SNR
06366 034200         LDA 3,C60
06367 054424         STA 3,ZSUPP      ;C(0)=DIGIT
06370 163000         ADD 3,0          ;MAKE ASCII
06371 175004         MOV 3,3,SZR
06372 004423         JSR CHAR.        ;PRINT
06373 000754         JMP DECOCT       ;GET NEXT DIGIT


06374 030426 OCTAB:  LDA 2,.+1+.-DECOCT
06375 100000         100000
06376 010000         10000
06377 001000         1000
06400 000100         100
06401 000010         10
06402 000001         1
06403 000000         0

06404 030436 DECTB:  LDA 2,.+1+.-DECOCT
      000012 .RDX 10
06405 023420         10000
06406 001750         1000
06407 000144         100
06410 000012         10
06411 000001         1
06412 000000         0
      000010 .RDX 8

06413 000000 ZSUPP:  0
06414 000000 RADRET: 0
```

```
▲ 0000  .MAIN

 06415 054454 CHAR.:  STA 3,CHRET      ;PRINT C(0) RIGHT
 06416 101325         MOVZS 0,0,SNR    ;RETURN +2 IF NULL
 06417 001401         JMP 1,3
 06420 040452         STA 0,CHSAV
 06421 176000         ADC 3,3          ;COMPUTE THE PARITY
 06422 117000         ADD 0,3
 06423 163404         AND 3,0,SZR
 06424 000775         JMP .-3
 06425 176660         SUBCR 3,3        ;COMBIND PARITY WITH CHAR
 06426 020444         LDA 0,CHSAV
 06427 163300         ADDS 3,0

 06430 034440 CHAR1:  LDA 3,CHTAB      ;IS THIS A TAB
 06431 116415         SUB# 0,3,SNR
 06432 000407         JMP .+7          ;YES
 06433 004446         JSR TYPE         ;NO PRINT IT
 06434 000413         JMP CHAR2+1      ;EXIT

 06435 020436         LDA 0,CHORZ      ;SIMULATE A TAB
 06436 034432         LDA 3,CHTAB      ;VIA 1 TO 9 SPACES
 06437 162426         SUBZ 3,0,SEZ
 06440 000777         JMP .-1
 06441 101005         MOV 0,0,SNR
 06442 000404         JMP CHAR2
 06443 020431         LDA 0,CH240
 06444 004435         JSR TYPE
 06445 000770         JMP .-10
 06446 040425 CHAR2:  STA 0,CHORZ
 06447 063511         SKPBZ TTO
 06450 000777         JMP .-1
 06451 060211         NIOC TTO
 06452 002417         JMP @CHRET
```

```
06453 060477 CRLF:   READS 0
06454 103102         ADDL 0,0,SZC      ;NO PRINT IF SW1=1
06455 001400         JMP 0,3
06456 054417         STA 3,CRLFR       ;SAVE RETURN
06457 020417         LDA 0,C215
06460 004735         JSR CHAR.         ;PRINT CARRIAGE AND LF
06461 020416         LDA 0,C212
06462 004733         JSR CHAR.
06463 020125         LDA 0,C377        ;PRINT RUB
06464 004731         JSR CHAR.
06465 102400         SUB 0,0
06466 040405         STA 0,CHORZ       ;CLEAR HORZ POSISTION
06467 002406         JMP @CRLFR        ;EXIT

06470 000011 CHTAB:  11
06471 000000 CHRET:  0
06472 000000 CHSAV:  0
06473 000000 CHORZ:  0
06474 000240 CH240:  240
06475 000000 CRLFR:  0
06476 000215 C215:   215
06477 000212 C212:   212

06500 000000 MESSR:  0
06501 054406 TYPE:   STA 3,TYPRET      ;TYPE THE C(0)R IF
06502 010771         ISZ CHORZ
06503 063511         SKPBZ TTO
06504 000777         JMP .-1
06505 061111         DOAS 0,TTO
06506 002401         JMP @TYPRET
06507 000000 TYPRET: 0
```

```
                        ;TYPE   0 FOR CARTRIDGE
                MSG1:   .TXTE |TYPE     0 FOR CARTRIDGE|
006510 054724
006511 142520
006512 030011
006513 143240
006514 151317
006515 141640
006516 151101
006517 151324
006520 042311
006521 142507
006522 000000
                        ;       1 FOR 10 SURFACE DISK PACK
                MSG2:   .TXTE | 1 FOR 10 SURFACE DISK PACK|
006523 130411
006524 143240
006525 151317
006526 130540
006527 120060
006530 052523
006531 143322
006532 141501
006533 120305
006534 144504
006535 045523
006536 050240
006537 141501
006540 000113
                        ;       2 FOR 20 SURFACE DISK PACK
                MSG3:   .TXTE | 2 FOR 20 SURFACE DISK PACK|
006541 131011
006542 143240
006543 151317
006544 131240
006545 120060
006546 052523
006547 143322
006550 141501
006551 120305
006552 144504
006553 045523
006554 050240
006555 141501
006556 000113
                        ; -?-
                MSG4:   .TXTE | -?-|
006557 026640
006560 026477
006561 000000
                        ;TYPE UNIT NUMBERS (0-3) TO TEST
                MSG5:   .TXTE |TYPE UNIT NUMBERS (0-3) TO TEST|
006562 054724
006563 142520
006564 052540
006565 144516
006566 120324
006567 052516
```

```
0093   .MAIN
 06570 041115
 06571 151305
 06572 120123
 06573 030050
 06574 031455
 06575 120251
 06576 147724
 06577 152240
 06600 051705
 06601 000324
                        !PASS
                MSG6!   .TXTE !PASS!
 06602 040520
 06603 051523
 06604 000000


 06605 000000 PRGEND!  0

       006700           .LOC 6700
                        .TXTE  !DKP DIAG 1!
 06700 045504
 06701 120120
 06702 144504
 06703 043501
 06704 130640
 06705 000000
 06706 000002           000002
 06707 000222           BEGIN
 06710 000002           000002
 06711 000000           000000
 06712 000020           000020
 06713 000000           000000
 06714 000000           000000
 06715 100033           100033

                        .END
```

MOVING HEAD DISC RELIABILITY

```
0001  .MAIN
01
02
03
04                ;       MOVING HEAD DISK RELIABILITY PROGRAM
05
06                ;1.     ABSTRACT
07
08                ;       THE MOVING HEAD DISK RELIABILITY PROGRAM
09                ;       IS A MAINTENANCE PROGRAM DESIGNED TO
10                ;       EXERCISE AND TEST THE 4040 DISK CONTROLLER
11                ;       AND 1-4 DISK DRIVES.  THE DISK DRIVES MAY BE
12                ;       SHARED BETWEEN TWO COMPUTERS IN WHICH CASE
13                ;       THIS PROGRAM MAY BE RUNNING IN EACH COMPUTER.
14
15                ;               NOTE
16                ;               ****
17                ;       ONE COMPUTER RUNNING RELIABILITY
18                ;       TEST (SA 401), THE OTHER (SA 402)
19                ;       ONLY!!!
20
21                ;       THE CONTROL CAN BE DEVICE 33 OR 73.
22
23                ;2.     MACHINE REQUIREMENTS
24
25                ;       NOVA FAMILY CENTRAL PROCESSOR
26                ;       4K READ/WRITE MEMORY
27                ;       TELETYPE
28                ;       4040 DISK CONTROL
29                ;       4047 OR 4040 CONTROL ADAPTER
30                ;       1-4 MOVING HEAD DISK DRIVES
31
32                ;3.     OPERATING PROCEDURES
33
34                ;       A. LOAD PROGRAM USING BINARY LOADER
35                ;       B. RESET AND START AT ONE OF THE
36                ;          ADDRESSES SHOWN BELOW.
37
38                ;       STARTING ADDRESS
39                ;         2     RUN ALL
40                ;         4     SET DISK CONTROL ADDRESS TO 33
41                ;         5     SET DISK CONTROL ADDRESS TO 73
42                ;       400     RELIABILITY TEST, ALL CYLINDERS
43                ;       401     RELIABILITY TEST, EVEN CYLINDERS
44                ;       402     RELIABILITY TEST, ODD CYLINDERS
45                ;       403     DISK ADDRESS TEST
46                ;       404     NOT USED
47                ;       405     COMMAND STRING INTERPRETER
48                ;       406     ONES TEST
49                ;       407     ZEROS TEST
50                ;       410     1101101101 TEST
51                ;       411     FLOATING ONE TEST
52                ;       412     FLOATING ZERO TEST
53                ;       413     RUN ALL
54                ;       414     RECALIBRATE UNIT 0
55                ;       415     RECALIBRATE UNIT 1
56                ;       416     RECALIBRATE UNIT 2
57                ;       417     RECALIBRATE UNIT 3
```

```
A 0002  .MAIN
01
02                ;       C. ERRORS - ERROR STATUS IS PRINTED
03                ;       WHENEVER ENCOUNTERED.  WHEN DATA ERRORS
04                ;       ARE FOUND ONLY THREE ARE PRINTED PER
05                ;       ENCOUNTER. (SEE PARAGRAPH 5)
06
07                ;       SWITCH SETTINGS
08                ;       SW0=1 DO NOT HALT FOLLOWING ERROR
09                ;       SW1=1 INHIBIT ERROR PRINTOUTS
10                ;       SW2=1 REPEAT CURRENT TEST (SCOPE LOOP)
11                ;       SW3=1 INHIBIT CHECK WORD AND DATA ERROR MESSAGES
12                ;       SW4=1 BREAK TO ALLOW DISK INTERCHANGE
13                ;       SW5=1 FOR READ ONLY MODE
14
15                ;       D. STATISTICS - TYPE ANY KEY DURING
16                ;       RANDOM TESTING TO GET A REPORT OF THE
17                ;       NUMBER OF WORDS WRITTEN AND READ, PLUS
18                ;       THE NUMBER OF ERRORS.
19
20                ;4.     PROGRAM DESCRIPTION
21
22                ;       A. RELIABILITY TEST (SA 400)
23                ;       A RANDOM NUMBER GENERATOR IS USED TO SELECT A
24                ;       DISK DRIVE,CYLINDER,HEAD,BEGINNING SECTOR,
25                ;       AND NUMBER OF CONSECUTIVE SECTORS.  RANDOM
26                ;       DATA IS THEN GENERATED, WRITTEN, AND READ.
27                ;       THE SEQUENCE IS REPEATED INDEFINITELY.
28
29                ;       B. RELIABILITY TEST (SA 401)
30                ;       SAME AS A, EXCEPT THAT ONLY EVEN
31                ;       NUMBERED CYLINDERS ARE USED.  THIS
32                ;       ALLOWS A TWO COMPUTER SYSTEM TO RUN
33                ;       SIMULTANEOUS RELIABILITY TESTS.
34
35                ;       C. RELIABILITY TEST (SA 402)
36                ;       SAME AS B, ONLY FOR ODD NUMBERED
37                ;       CYLINDERS.
38
39                ;       D. DISK ADDRESS TEST (SA 403)
40                ;       RANDOM DATA IS FIRST WRITTEN THEN READ
41                ;       FROM ALL SECTORS ON EACH READY DISK.  THIS
42                ;       INSURES THAT ALL DISK PACK SURFACES ARE
43                ;       USEABLE AND THAT THE DISK PACK IS FORMATTED
44                ;       PROPERLY.
```

```
A 0003  .MAIN
01
02
03              !       F. COMMAND STRING INTERPRETER (SA 403)
04              !       AS A TROUBLE SHOOTING AID THE SERVICE
05              !       ENGINEER MAY TYPE IN HIS OWN TEST LOOP.
06              !       AFTER STARTING AT 400, THREE ARGUMENTS
07              !       MUST BE ENTERED IN RESPONSE TO THREE
08              !       PROGRAM QUESTIONS; "UNIT", "DATA", AND
09              !       "COMMAND STRING".
10
11
12
13              !       I.      UNIT:   TYPE UNIT # OR CARRIAGE TO
14              !                       USE THE PREVIOUS ENTRY
15
16              !       II.     DATA:   RAN=RANDOM
17              !                       ALO=ALL ONES
18              !                       ALZ=ALL ZEROS
19              !                       PAT=110110 PATTERN
20              !                       FLO=FLOATING ONE PATTERN
21              !                       FLZ=FLOATING ZERO PATTERN
22
23              !                       ALTERNATIVELY ENTER A STRING OF
24              !       1.              OCTAL 18 BIT WORDS TO BE
25              !                       USED AS DATA.  THE WORDS
26              !                       ENTERED ARE USED REPEATEDLY
27              !                       TO MAKE UP A SECTOR BLOCK.
28              !                       TYPE CARRIAGE TO USE THE
29              !                       PREVIOUS ENTRY.
30
31              !       III.    COMMAND STRING:
32
33              !   1.  OPTIONS 1.      READ HEAD,SECTOR,#SECTORS
34              !               2.      WRITE SAME
35              !               3.      SEEK CYLINDER
36              !               4.      RECALIBRATE
37              !               5.      LOOP (GO TO BEGINNING)
38              !               6.      DELAY (N)  12.5MS INCREMENTS
39              !               7.      TYPE CARRIAGE TO USE THE
40              !                       PREVIOUS COMMAND STRING.
41
42              !               NOTE THAT EITHER SPACES OR A COMMA
43              !               MAY BE USED AS AN ARGUMENT DELIMITER.
44              !               EACH RESPONSE IS TERMINATED BY
45              !               TYPING CARRIAGE RETURN.  IF MORE
46              !               ROOM IS NEEDED ON A LINE, TYPE
47              !               LINE FEED TO SPACE TO THE NEXT LINE.
48              !               THE WORD "SAME" USED WITH READ, OR WRITE,
49              !               WILL CAUSE THE PREVIOUS DISK
50              !               ADDRESS PARAMETERS TO BE USED.
```

```
A 0004  .MAIN
01
02                ;       THE FOLLOWING EXAMPLE WOULD CAUSE UNIT
03                ;       1 TO REPEATEDLY SEEK  CYLINDER
04                ;       00, WRITE SECTORS 2 AND 3 WITH HEAD 5,
05                ;       THEN READ IT BACK AND CHECK.  DATA IS SPECIFIED
06                ;       AS ALTERNATE WORDS OF ZEROS THEN ONES.
07
08                ;       UNIT: 1
09                ;       DATA: 0,177777
10                ;       COMMAND STRING: SEEK 50 WRITE 5,2,2 READ SAME LOOP
11
12
13                ;   1.  G, ONES TEST  (DATA = ALL 1'S)
14                ;          ZEROS TEST  (DATA = ALL 0'S)
15                ;          110110 TEST  (DATA = WORDS OF 1101101101101101)
16                ;          FLOATING ONE TEST  (EACH SUCCESSIVE WORD
17                ;               CONTAINS ONE 1 BIT WHICH IS MOVED
18                ;               RIGHT ONE BIT EACH WORD)
19                ;          FLOATING ZERO TEST  (COMPLEMENT OF THE
20                ;               FLOATING 1 TEST)
21
22                ;       EACH OF THE ABOVE USE THE BASIC DISK ADDRESS
23                ;       TEST, SUBSTITUTING THE APPROPRIATE DATA.
24
25                ;       ANY OF THESE PATTERNS INCLUDING THE RANDOM
26                ;       DATA USED FOR THE DISK ADDRESS TEST MAY
27                ;       BE RUN IN THE "READ ONLY" MODE.  THIS IS USEFUL
28                ;       FOR CHECKING THE INTERCHANGEABILITY OF DISK
29                ;       PACKS BETWEEN VARIOUS DISK DRIVES.
30
31                ;       TO GENERATE A DATA PATTERN START AT THE
32                ;       APPROPRIATE LOCATION AND SET SW4 TO A 1.
33                ;       WHEN THE ENTIRE PACK HAS BEEN WRITTEN AND
34                ;       READ THE TELETYPE PRINTS "INTERCHANGE DISK"
35                ;       AND THE PROGRAM HALTS.  THIS PACK MAY NOW
36                ;       BE READ FROM OTHER DISK DRIVES IN THE READ
37                ;   2.  ONLY MODE (SW8=1).
```

```
A 0005  .MAIN
01
02              !5. ERROR REPORTING AND RECOVERY
03
04              !       ALL PHASES OF THE SOFTWARE WORK THROUGH
05              !     1 4 MAIN SUBROUTINES DESCRIBED BELOW.  EACH
06              !       SUBROUTINE HAS A NORMAL RETURN (+3) AND
07              !       AN ERROR RETURN (+1). EACH SUBROUTINE WAITS FOR DISK
08              !       COMPLETION WITH INTERRUPT ENABLED.  A FAILURE
09              !       TO DETECT INTERRUPT WITHIN 500MS (3 SEC FOR
10              !       RECALIBRATE) RESULTS IN A "TIMEOUT" ERROR.
11
12              !       RECALIBRATE - ANY UNUSUAL STATUS IS REPORTED
13              !       IMMEDIATELY AND AN ERROR RETURN EXECUTED.
14
15              !       SEEK - SEEK ERROR STATUS INCREMENTS SEEK
16              !       ERROR COUNTER.  ANY ERROR STATUS  RESULTS
17              !       IN STATUS PRINTOUT.
18
19              !       WRITE - FOLLOWING "DONE" ON A WRITE, ERRORS ARE
20              !       CHECKED IN THE SEQUENCE SHOWN BELOW.  ERROR
21              !       RECOVERY PROCEDURE IS OUTLINED FOR EACH CASE.
22              !       IF THE ERROR IS NOT PRESENT THE NEXT CHECK IS MADE.
23
24              !        1. READ/WRITE DONE STATUS - IF NONE, INCREMENT THE
25              !        MISC ERROR COUNT, PRINT ILLEGAL STATUS MESSAGE
26              !        AND DO AN ERROR RETURN.
27
28              !        2. MISC STATUS BITS - (ANY SEEK DONE, ANY SEEKING
29              !        BIT, SEEK ERROR, END CYLINDER, OR DATA LATE).
30              !        IF ANY ERROR INCREMENT THE MISC ERROR COUNT,
31              !        PRINT THE ILLEGAL STATUS, AND DO AN ERROR RETURN.
32
33              !        3. ADDRESS ERROR
34              !               3.1 FIRST TIME - INCREMENT ADDRESS ERROR COUNT
35              !               AND REPEAT THE WRITE.
36              !               3.2 SECOND SUCCESSIVE FAILURE - INCREMENT
37              !               PERMANENT ADDRESS ERROR COUNT AND DO A
38              !               ERROR RETURN.
39
40              !        4. ENDING MEMORY ADDRESS - INCREMENT THE MISC ERROR
41              !        COUNTER, PRINT THE ERROR MESSAGE, SET THE FATAL FLAG,
42              !        AND GO TO 5.
43
44              !        5. ENDING DISK ADDRESS - INCREMENT THE MISC ERROR
45              !        COUNTER, PRINT THE ERROR MESSAGE, SET THE FATAL FLAG,
46              !        AND GO TO 6.
47
48              !        6. NO FURTHER CHECKS
49              !               6.1 FATAL SWITCH ON - DO A ERROR RETURN.
50              !               6.2 OTHERWISE - DO A NORMAL RETURN.
```

```
0006  .MAIN
00
02                    !       READ - FOLLOWING "DONE" ON A READ, ERRORS ARE
03                    !       CHECKED IN THE SEQUENCE SHOWN BELOW.  ERROR
04                    !       RECOVERY PROCEDURE IS OUTLINED FOR EACH CASE.
05                    !       IF THE ERROR IS NOT PRESENT THE NEXT CHECK IS MADE.
06
07                    !        1. READ/WRITE DONE STATUS - IF NONE, INCREMENT THE
08                    !        MISC ERROR COUNT, PRINT ILLEGAL STATUS MESSAGE
09                    !        AND DO AN ERROR RETURN.
10
11                    !        2. MISC STATUS BITS - (ANY SEEK DONE, ANY SEEKING
12                    !        BIT, SEEK ERROR, END CYLINDER, OR DATA LATE).
13                    !        IF ANY ERROR INCREMENT THE MISC ERROR COUNT,
14                    !        PRINT THE ILLEGAL STATUS, AND DO AN ERROR RETURN.
15
16                    !        3. ADDRESS ERROR
17                    !              3.1 FIRST TIME - INCREMENT ADDRESS ERROR COUNT
18                    !              AND REPEAT THE READ.
19                    !              3.2 SECOND SUCCESSIVE FAILURE - INCREMENT
20                    !              PERMANENT ADDRESS ERROR COUNT AND DO A
21                    !              ERROR RETURN.
22
23                    !        4. CHECK WORD ERROR
24                    !              4.1 FIRST TIME - INCREMENT THE CHECK WORD
25                    !              ERROR COUNT AND SET THE RETRY FLAG.  PRINT
26                    !              "CHECK WORD ERROR" AND GO TO 5.
27                    !              4.2 SECOND SUCCESSIVE ERROR - INCREMENT THE
28                    !              PERMANENT CHECK WORD ERROR COUNTER AND SET
29                    !              THE FATAL FLAG.  PRINT "CHECK WORD ERROR"
30                    !              AND GO TO 5.
31
32                    !        5. DATA ERROR
33                    !              5.1 FIRST TIME - SET THE RETRY FLAG AND PRINT
34                    !              ERROR REPORT.
35                    !                     5.1.1 CHECK WORD ERROR - DECREMENT THE
36                    !                     CHECK WORD ERROR COUNTER AND INCREMENT
37                    !                     THE CHECK WORD & DATA ERROR COUNTER.
38                    !                     GO TO 6.
39                    !                     5.1.2 NO CHECK WORD ERROR - INCREMENT
40                    !                     THE DATA ERROR COUNTER AND GO TO 6.
41                    !              5.2 SECOND SUCCESSIVE ERROR - SET THE FATAL
42                    !              FLAG AND PRINT THE ERROR REPORT.
43                    !                     5.2.1 CHECK WORD ERROR - DECREMENT
44                    !                     THE PERMANENT CHECK WORD ERROR COUNTER
45                    !                     AND INCREMENT THE PERMANENT CHECK WORD
46                    !                     & DATA ERROR COUNTER.  GO TO 6.
47                    !                     5.2.2 NO CHECK WORD ERROR - INCREMENT
48                    !                     THE PERMANENT DATA ERROR COUNTER.
49                    !                     GO TO 6.
```

```
A 0007   .MAIN
01
02                ;        5. ENDING MEMORY ADDRESS = INCREMENT THE MISC ERROR
03                ;        COUNTER, PRINT THE ERROR MESSAGE, SET THE FATAL FLAG
04                ;        AND GO TO 7.
05
06                ;        7. ENDING DISK ADDRESS = INCREMENT THE MISC ERROR
07                ;        COUNTER, PRINT THE ERROR MESSAGE, SET THE FATAL FLAG
08                ;        AND GO TO 8.
09
10                ;        8. NO FURTHER CHECKS
11                ;               8.1 RETRY SWITCH ON = PRINT "TRY AGAIN"
12                ;               AND REPEAT THE TEST.
13                ;               8.2 FATAL SWITCH ON = DO A ERROR RETURN.
14                ;               8.3 NO SWITCHES ON = DO A NORMAL RETURN.
```

```
A 0008   .MAIN
01
02
03          000001              .LOC 1
04  00001  000070               INTERRUPT
05  00002  002003               JMP 0,.+1          ;START HERE, RUN ALL
06  00003  000413               .RAL
07  00004  002006               JMP 0,.+2          ;SET ADDR TO 33
08  00005  002007               JMP 0,.+2          ;SET ADDR TO 73
09  00006  000520               FXADD
10  00007  000521               FXADD+1
11
12          000022              .LOC 22
13
14  00022  000001  DRVS:        1                  ;UNIT 0 CODE
15  00023  000002               2                  ;      1
16  00024  000004               4                  ;      2
17  00025  000010               10                 ;      3
18
19  00026  000000  UNTINS:      0
20  00027  040000               40000
21  00030  100000               100000
22  00031  140000               140000
23  00032  040000  UNTON:       40000
24  00033  020000               20000
25  00034  010000               10000
26  00035  004000               4000
27  00036  000000  FLO1:        0
28  00037  000000  FLOZ:        0
29
30          000041              .LOC 41
31  00041  003740               TYPE
```

```
A 0009   .MAIN
01
02          000045              .LOC 45
03          000004 CYLF:        .BLK 4          ;FROM CYLINDERS
04          000004 CYLT:        .BLK 4          ;TO CYLINDERS
05          000010 SEEKT:       .BLK 10         ;TOTAL SEEKS
06          000004 SEKER:       .BLK 4          ;TOTAL SEEK ERRORS
07          000010 WDSW:        .BLK 10         ;WORDS WRITTEN
08          000010 WDSR:        .BLK 10         ;WORDS READ
09          000004 CWER:        .BLK 4          ;CHECK WORD ERRORS
10          000004 CWERP:       .BLK 4          ;PERM CHECK WORD ERRORS
11          000004 ADDER:       .BLK 4          ;ADDRESS ERRORS
12          000004 PADER:       .BLK 4          ;PERMANENT ADDR ERR
13          000004 DATER:       .BLK 4          ;DATA ERRORS
14          000004 CWDE:        .BLK 4          ;CHK WD AND DATA ERROR
15          000004 CWDEP:       .BLK 4          ;PERM CHK WD AND DATA ERROR
16          000004 PDER:        .BLK 4          ;PERMANENT DATA ERRORS
17          000004 MISC:        .BLK 4          ;MISC ERRORS
18
19  00155 000000 HSW:      0                   ;HEADER SWITCH
20  00156 000000 CSIF:     0                   ;CMD STRING INTRP FLAG
21  00157 000000 UNIT:     0                   ;CURRENT DISK DRIVE UNIT
22  00160 000000 RDYUNT:   0                   ;AVAILABLE UNITS
23  00161 000000 RALL:     0                   ;RUN ALL MODE FLAG
24  00162 000000 RETRY:    0                   ;RETRY TEST FLAG
25  00163 000000 FATAL:    0                   ;FATAL TEST RESULT FLAG
26  00164 000000 MODE:     0                   ;0=RECAL, 1=SEEK, 2=READ, 3=WRITE
27  00165 000000 LUPSW:    0                   ;LOOP ON TEST SWITCH
28  00166 000000 SSW:      0                   ;SEEK SWITCH
29  00167 000167           .
30
31  00170 000000 LHD:      0                   ;LAST HEAD+1)  2,10,OR 20.
32  00171 000000 LS:       0                   ;LAST SECTOR+1)  6 OR 12
33
34  00172 002000 UNTSK:    2000
35  00173 001000           1000
36  00174 000400           400
37  00175 000200           200
```

```
▲ 0010   .MAIN
01
02          000023 C2=DRVS+1
03 00176    000003 C3:      3
04          000024 C4=DRVS+2
05 00177    000007 C7:      7
06          000025 C10=DRVS+3
07 00200    000012 C12:     12
08 00201    000017 C17:     17
09 00202    000020 C20:     20
10 00203    000033 C33:     33
11 00204    000037 C37:     37
12 00205    000040 C40:     40
13 00206    000054 C54:     54
14 00207    000060 C60:     60
15 00210    000067 C67:     67
16 00211    000100 C100:    100
17 00212    000177 C177:    177
18 00213    000313 C203.:   203.
19 00214    000212 C212:    212
20 00215    000215 C215:    215
21 00216    000377 C377:    377
22          000174 C400=UNTSK+2
23          000173 C1000=UNTSK+1
24 00217    001400 C1400:   1400
25          000172 C2000=UNTSK
26          000035 C4K=UNTDN+3
27 00220    007400 C7400:   7400
28 00221    017400 C174H:   174000
29          000034 C10K=UNTDN+2
30 00222    170000 C170K:   170000
31 00223    176000 C176K:   176000
32 00224    077662 CSP:     77662
33 00225    177677 CSP1:    177677
34 00226    060033 CWRD:    60033
35 00227    077666 CSP3:    77666
36
37
38
39
40
41
42
43
44
45 00230    177760 M16.:    -16.
46 00231    177546 M100.:   -100.
47 00232    177426 M250.:   -250.
48 00233    177737 M41:     -41
```

```
A 0011   .MAIN
01
02 00234 000000 TIME:    0              ;# IOT'S FOR 12.5MS ( THIS COMPUTER )
03 00235 000000 TIMER:   0              ;12.5MS TIMER
04 00236 000000 IOTAT:   0              ;STATUS AT INTERRUPT
05 00237 000000 SAV0:    0              ;SAVE AC0,1,2
06 00240 000000 SAV1:    0
07 00241 000000 SAV2:    0
08 00242 000000 SAV3:    0
09 00243 123456 RANDOM:  123456         ;CURRENT RANDOM NUMBER
10 00244 123456 .RAN:    123456         ;RANDOM BASE NUMBER
11 00245 000000 RELRAN:  0
12 00246 060200 CNIOC:   NIOC 0
13 00247 005150 LAST:    PRGEND         ;PROGRAM END
14 00250 000000 LINCT:   0              ;BYTE POINTER
15 00251 000000 SEC:     0              ;STARTING SECTOR (POS)
16 00252 000000 SC:      0              ;SECTOR COUNT
17 00253 000000 HD:      0              ;HEAD
18 00254 005150 CA:      PRGEND         ;BEGINNING ADDRESS (VARIABLE)
19 00255 040055 SAM:     40055          ;"SAME
20 00256 000000 TERM:    0              ;HOLDS TERMINATOR FLAG
21 00257 011710 UDP:     UDUFF+UBUFF    ;UNIT BYTE POINTER
22 00260 012074 CCOP:    CDUFF+CBUFF    ;CMD STRING BYTE POINTER
23 00261 005127 VARST:   VAR-1          ;VARIABLE DATA, SRT ADDR-1
24 00262 000147 VARED:   VAR+15.        ;     "       "   TABLE END
25 00263 000000 VARSP:   0              ;     "       "   END MARKER
26 00264 000000 VARPT:   0              ;VAR POINTER
27 00265 005150 BUFF:    PRGEND         ;DATA BUFFER SRT ADDR (CONSTANT)
28 00266 014000 CMEND:   14000          ;TOP OF USABLE CORE
29 00267 000000 SMAX:    0              ;MAX SECTOR COUNT
30 00270 000000 EVODN:   0              ;CYLINDERS, (0)EVEN, (15)ODD
31 00271 000000 AECNT:   0
32 00272 000000 CWCNT:   0
33 00273 000000 DACNT:   0
34 00274 000000 RWRET:   0
35 00275 000000 DADAT:   0              ;DATA ADDRESS (DISK ADDR TEST)
36 00276 000000 CFLG:    0              ;END CYL FLAG
37 00277 000033 CDSK:    33
38 00300 000000 COC:     0
39 00301 000400 FIRST:   BEGIN
40 00302 000000 UMSK:    0              ;UNIT MASK 1400 OR 400 OR 0
41 00303 000000 HMSK:    0              ;HEAD MASK 7400 OR 17400 OR 400
42 00304 000000 SMSK:    0              ;SECTOR MASK 7400 OR 3400
43 00305 003400 LMSK:    3400           ;# SECTORS MASK 7400 OR 3400
44 00306 000000 .SC:     0              ;# SECTOR TRANSFERRED ON CHK WD ERR
45 00307 000000 RLUP:    0              ;REPEAT TEST ADDRESS
46 00310 000000 ALLRET:  0
```

```
A 0012  .MAIN
01
02 00311 000577 ISET.:  SET.1
03 00312 000070 IRUP:   INTERRUPT
04 00313 000045 ICYLF:  CYLF
05 00314 001234 IALL:   ALL
06 00315 001256 IDLY:   DLY
07 00316 003752 ISAC:   SAC
08 00317 003756 ISTAC:  STAC
09 00320 003600 IPDEC:  PDEC
10 00321 003575 ITAC1:  POCT
11 00322 003574 ITZ1:   ZOCT
12 00323 003551 IMESS:  MESS
13 00324 003714 ICRLF:  CRLF
14 00325 003010 ISRH:   SRH
15 00326 003006 IGATM:  GATM
16 00327 002737 IHSS:   HSS
17 00330 003762 IINP:   INP
18 00331 003241 IGEN:   G
19 00332 003240 ICHK:   C
20 00333 003363 IVAR:   VAR.0
21 00334 001016 ICLRB:  CLRB
22 00335 001030 ISET:   SET
23 00336 003020 ISM:    SM
24 00337 000660 ISTB:   STB
25 00340 000612 INIL:   INIL.
26 00341 000726 WAT:    WAIT+1
27 00342 003434 I.DBD:  .DDD
28 00343 002647 IRDAT:  RDATA
29 00344 002325 IWDAT:  WDATA
30 00345 002105 IGCS:   GCS
31 00346 002611 IR:     .READ
32 00347 002310 IW:     .WRITE
33 00350 002253 IS:     .SEEK
34 00351 002223 IRC:    RECL
35 00352 001265 IPS:    PS
36 00353 003170 IHED:   HED
37 00354 002774 ICSW:   .CSW
38 00355 001747 ISCNT:  SCNT
39 00356 002003 ICSI:   CMDST
40 00357 002001 IQUST:  QUEST
41 00360 000725 IWT:    WAIT
42 00361 003377 IRAN:   RAN
43 00362 001274 I.GSD:  .GSD
44 00363 002022 ISU:    .SET
45 00364 000582 IRCAL:  RCALL
```

```
A 0013  .MAIN
01
02          000020 IDX0=20
03          000021 IDX1=21
04          000033 .DSKP=33
05          000314 DOALL=JSR @IALL          ;DO FOLLOWING ROUTINE FOR ALL RDY UNITS
06          000315 DLY12=JSR @IDLY          ;DELAY 12.0MS
07          000301 RAND=JSR @IRAN           ;GENERATE RANDOM # IN AC0
08          000310 SAVAC=JSR @ISAC          ;SAVE AC0,1,2
09          000317 GETAC=JSR @ISTAC         ;RESTORE AC0,1,2
10          000320 TYPDEC=JSR @IPDEC        ;TYPE (AC1) IN DECIMAL
11          000321 TYPAC1=JSR @ITAC1        ;TYPE (AC1) IN OCTAL
12          000322 TYPZ1=JSR @ITZ1          ;TYPE (AC1) IN OCTAL, SUPP LEAD 0'S
13          000323 MESSAGE=JSR @IMESS       ;TYPE FOLLOWING MESSAGE IN ASCII
14          000324 PCRLF=JSR @ICRLF         ;TYPE CR=LF
15          000330 INPUT=JSR @IINP          ;GET A LINE OF INPUT
16          000332 CHECK=JSR @ICHK          ;CHECK ALL OF DATA BUFFER
17          000331 GEN=JSR @IGEN            ;GENERATE DATA BUFFER
18          000327 GETPAR=JSR @IHSS         ;GET HEAD=SECT=# SECT FROM INPUT LINE
19          000326 GETATM=JSR @IGATM        ;GET NAME OR # FROM INPUT LINE
20          000325 SEARCH=JSR @ISRH         ;SEARCH FOLLOWING TABLE FOR MATCH ON AC1
21          000334 CLRRD=JSR @ICLRB         ;CLEAR READ BUFFER
22          000335 SETP=JSR @ISET           ;SET DISK PARAMETERS
23          000336 SMEM=JSR @ISM            ;SIZE MEMORY
24          000337 SETD=JSR @IS7D           ;SET TIME BASE
25          000340 INITE=JSR @IINIL         ;INITIALIZE ERROR COUNTERS ETC.
26          002352 POTAT=JMP @IPS           ;PRINT STATUS
27          000346 READ=JSR @IR             ;DISK READ & CHECK DATA
28          000347 WRITE=JSR @IW            ;GENERATE DATA AND WRITE DISK
29          000350 SEEK=JSR @IS             ;SEEK NEW CYLINDER
30          000351 RECAL=JSR @IRC           ;RECALIBRATE
31          000354 CKSW=JSR @ICSW           ;CHECK CONSOLE SWITCHES
32          000353 HEADER=JSR @IHED         ;PRINT ERROR MESSAGE HEADER
33          000300 INTWT=JSR @IWT           ;INTERRUPT WAIT ROUTINE
34          006041 TYPASC=JSR @41           ;TYPE (AC0)R IN ASCII
35
36          006363 SETU=JSR @ISU            ;SET READY UNITS
37
38                      .EOT
```

```
 0014  .MAIN
01                ;       ********************************************
02                ;       **      STARTING ADDRESSES                ***
03                ;       ********************************************
04
05        000400          .LOC 400
06
07 00400 004420 BEGIN:    JSR STRT          ;RELIABILITY TEST, ALL CYL.
08 00401 004417           JSR STRT          ;       "          EVEN  "
09 00402 004416           JSR STRT          ;       "           ODD  "
10 00403 004413           JSR STRT          ;DISK ADDRESS TEST
11 00404 063077           HALT              ;NOT USED
12 00405 000464           JMP STRC          ;COMMAND STRING INTERPRETER
13 00406 004412           JSR STRT          ;ONES TEST
14 00407 004411           JSR STRT          ;ZEROS TEST
15 00410 004410           JSR STRT          ;110110 TEST
16 00411 004407           JSR STRT          ;FLOATING ONES TEST
17 00412 004406           JSR STRT          ;FLOATING ZEROS TEST
18 00413 004405 .RAL:     JSR STRT          ;RUN ALL
19 00414 002364           JMP @IRCAL        ;RECALIBRATE UNIT 0
20 00415 002364           JMP @IRCAL        ;RECALIBRATE UNIT 1
21 00416 002364           JMP @IRCAL        ;RECALIBRATE UNIT 2
22 00417 002364           JMP @IRCAL        ;RECALIBRATE UNIT 3
23
24                ;       ********************************************
25                ;       ***     INITIALIZATION                    **
26                ;       ********************************************
27
20 00420 020301 STRT:     LDA 0,FIRST       ;INITIALIZATION, ALL
29 00421 116400           SUB 0,3
30 00422 054471           STA 3,INDEX
31 00423 006330 RSTRT:    SMEM              ;SIZE MEMORY
32 00424 006337           SETB              ;SET TIME BASE
33 00425 006340           INITE             ;INITIALIZE BUFFERS, COUNTERS, ETC.
34 00426 006303           SETU              ;SET READY UNITS
35 00427 120400           SUB 1,1           ;AC0=READY UNIT PATTERN
36 00430 101225           MOVZR 0,0,SNR
37 00431 000404           JMP RS3           ;UNIT 0 ONLY
38 00432 101224           MOVZR 0,0,SZR
39 00433 125140           MOVOL 1,1         ;UNIT 2 OR 3     MASK=3
40 00434 125140           MOVOL 1,1         ;UNIT 1          MASK=1
41 00435 125300 RS3:      MOVS 1,1
42 00436 044502           STA 1,UMSK        ;UNIT MASK
43 00437 063077           IORST
44 00440 020170           LDA 0,LHD         ;IO DISK SIZE
45 00441 101003           MOV 0,0,SNR       ;SPECIFIED ??
46 00442 000403           JSR @I.GSD        ;NO, GET IT
47 00443 000004           PCRLF
48 00444 000023           MSGDAGE
49 00445 000072           MSG72             ;"TESTING UNIT "
50 00446 000014           DOALL
51 00447 000407           TUN               ;TYPE UNIT NUMBERS
52 00450 063011           SKPBZ TTO         ;WAIT FOR TTO
53 00451 000777           JMP .-1           ;TO FINISH
54 00452 063211           NIOC TTO          ;CLEAR DONE
55 00453 022400           LDA 0,STR,2
56 00454 054457           LDA 2,INDEX
57 00455 113000           ADD 0,2
58 00456 003000           JMP @0,2          ;GO TO IT !!
```

```
A 0010  .MAIN
01
02 00457 004410 TUN:    STA 3,TUNRET     ;TYPE UNIT NUMBER
03 00460 000107          LDA 0,UNIT       ;FOLLOWED BY A ","
04 00461 024007          LDA 1,C60
05 00462 123000          ADD 1,0
06 00463 000041          TYPASC
07 00464 020464          LDA 0,COMA
08 00465 000041          TYPASC
09 00466 002401          JMP @TUNRET
10 00467 000000 TUNRET:  0
11 00470 000054 COMA:    ",
12
13                       ;COMMAND STRING INTERPRETER DOES NOT REQUIRE
14                       ;NORMAL INITIALIZATION.
15
16 00471 000336 STRC:    SMEM             ;SIZE MEMORY
17 00472 000337          SETB             ;SET TIME BASE
18 00473 000340          INITE            ;INITIALIZE BUFFERS ETC.
19 00474 000362          JSR @I.GSD       ;GET LAST HEAD/SECTOR
20 00475 002407          JMP @.CMD        ;DISPATCH TO ROUTINE
21
22
23                ;      ********************************************
24                ;      ***     DISPATCH TABLE TO PROGRAMS       **
25                ;      ********************************************
26
27 00476 000476 STR.2:   .
28 00477 001300          RELALL           ;RELIABILITY TEST, ALL CYL
29 00500 001340          RELEV            ;       "          EVEN  "
30 00501 001340          RELOD            ;       "          ODD   "
31 00502 001507          DATR             ;DISK ADDRESS TEST
32 00503 000000          0                ;NOT USED
33 00504 002003 .CMD:    CMDST            ;COMMAND STRING INTERPRETER
34 00505 001503          DAT1             ;ONES TEST
35 00506 001505          DAT0             ;ZEROS TEST
36 00507 001507          DATP             ;110110 PATTERN
37 00510 001571          DATF1            ;FLOAT ONE
38 00511 001573          DATF0            ;FLOAT ZERO
39 00512 001530          RUNALL           ;RUN ALL PARTS
40 00513 000000 INDEX:   0
```

```
▲ 0016  .MAIN
01
02 00514 000000 CSP2:   0
03 00515 100037 CIOMSK: 100037  ;*****************************************
04                              ;**      INITIALIZATION                   **
05                              ;**              SUBROUTINES              **
06 00516 002773 .LST:   HIADD   ;*****************************************
07 00517 000420 .LST1:  STRT
08
09                      ;CHANGE I/O ADDRESS TO 33 OR 73
10
11 00520 102401 FXADD:  SUB 0,0,SKP     ;MAKE IT 33
12 00521 020205         LDA 0,C40       ;MAKE IT 73
13 00522 040772         STA 0,CSP2
14 00523 030774         LDA 2,.LST1     ;POINTER IN MEMORY
15 00524 024771 FX.2:   LDA 1,CIOMSK    ;INST MASK
16 00525 021000         LDA 0,0,2       ;GET A WORD
17 00526 107400         AND 0,1
18 00527 034220         LDA 3,CWRD      ;IS IT AN I/O 33 OR 73 ?
19 00530 136404         SUB 1,3,SZR
20 00531 000406         JMP FX.3        ;NO
21 00532 024233         LDA 1,M41       ;MASK
22 00533 123400         AND 1,0         ;GET RID OF BIT 10
23 00534 024760         LDA 1,CSP2
24 00535 123000         ADD 1,0         ;MAKE ADDR 33 OR 73
25 00536 041000         STA 0,0,2       ;ADDR CHANGED
26 00537 151400 FX.3:   INC 2,2
27 00540 034750         LDA 3,.LST
28 00541 156404         SUB 2,3,SZR
29 00542 000762         JMP FX.2        ;MORE
30 00543 020203         LDA 0,C33       ;SET (CDSK) TO
31 00544 024750         LDA 1,CSP2      ;EQUAL THE DEVICE ADDR
32 00545 107000         ADD 0,1
33 00546 044277         STA 1,CDSK
34 00547 063077 FX.4:   HALT            ;ALL DONE
35
36                      ;RECALIBRATE THE UNIT IN SW 14-15
37
38 00550 060477 RCALL:  READS 0
39 00551 024176         LDA 1,C3
40 00552 123400         AND 1,0
41 00553 040157         STA 0,UNIT
42 00554 000001         RECAL           ;RECALIBRATE !
43 00555 000772         JMP FX.4
44 00556 000771         JMP FX.4
45 00557 000770         JMP FX.4
```

```
A 0017  .MAIN
01
02
03                              ;FIND COMPUTER TIME BASE
04
05 00500 062077 STB:    IORST
06 00501 054430         STA 3,STBRET
07 00502 020311         LDA 0,ISET,
08 00503 040001         STA 0,1             ;SET INTERRUPT RETURN
09 00504 020210         LDA 0,C377
10 00505 120400         SUB 1,1
11 00506 044234         STA 1,TIME
12 00507 061111         DOAS 0,TTO
13 00570 063011         SKPDN TTO
14 00571 000777         JMP .-1
15 00572 061111         DOAS 0,TTO
16 00573 060177         INTEN
17 00574 010234         ISZ TIME
18 00575 000777         JMP .-1
19 00576 063077         HALT                ;WAITED TOO LONG
20
21 00577 060211 SET.1:  NIOC TTO
22 00600 020234         LDA 0,TIME          ;TIME=100MS COUNT
23 00601 101220         MOVZR 0,0
24 00602 101220         MOVZR 0,0
25 00603 101220         MOVZR 0,0
26 00604 040234         STA 0,TIME          ;TIME=12.5MS COUNT
27 00605 062077         IORST
28 00606 020312         LDA 0,IRUP
29 00607 040001         STA 0,1
30 00610 002401         JMP @STBRET
31 00611 000000 STBRET: 0
32
33                              ;CLEAR ERROR COUNTERS ETC.
34
35 00612 030313 INIL.:  LDA 2,ICYLF
36 00613 024167         LDA 1,SSW+1
37 00614 102400         SUB 0,0
38 00615 041000         STA 0,0,2
39 00616 151400         INC 2,2
40 00617 132414         SUB# 1,2,SZR
41 00620 000775         JMP .-3
42 00621 061400         JMP 0,3
43
```

```
A 0010  .MAIN
01
02                          ;SET READY UNITS
03
04 00022 054443 .SET1:      STA 3,.SRET
05 00023 102400             SUB 0,0             ;DO A SEEK ON UNIT 0
06 00024 003033             DOC 0,.DSKP         ;TO SELECT THE ADAPTER
07 00025 020173             LDA 0,C1000
08 00026 061333             DOAP 0,.DSKP
09
10 00027 024172             LDA 1,C2000         ;WAIT FOR SEEKING
11 00030 060433             DIA 0,.DSKP         ;FLOP TO TURN ON
12 00031 123405             AND 1,0,SNR
13 00032 000776             JMP .-2
14
15 00033 020200             LDA 0,M16.          ;WAIT 200MS FOR
16 00034 000315             DLY12               ;THE SEEK 0 TO END
17 00035 101404             INC 0,0,SZR
18 00036 000776             JMP .-2
19
20 00037 152400             SUB 2,2
21 00040 050100             STA 2,RDYUNT        ;SET ALL UNITS "NOT READY"
22 00041 021020 .SET1:      LDA 0,UNTINS,2      ;SELECT UNITS 0-3
23 00042 003033             DOC 0,.DSKP         ;AND CHECK EACH
24 00043 060433             DIA 0,.DSKP         ;FOR "DUR"
25 00044 024211             LDA 1,C100
26 00045 123405             AND 1,0,SNR
27 00046 000405             JMP .SET2           ;NOT READY
28 00047 021022             LDA 0,DRVS,2        ;(RDYUNT) DEFINES WHICH
29 00050 024100             LDA 1,RDYUNT        ;UNITS ARE READY
30 00051 123000             ADD 1,0             ;BIT 15=UNIT 0, BIT 14=UNIT 1
31 00052 040160             STA 0,RDYUNT        ;BIT 13=UNIT 2, BIT 12=UNIT 3
32
33 00053 151400 .SET2:      INC 2,2
34 00054 024024             LDA 1,C4            ;EXIT AFTER ALL UNITS
35 00055 132414             SUB# 1,2,SZR        ; TESTED
36 00056 000763             JMP .SET1
37
38 00057 020100             LDA 0,RDYUNT
39 00060 101004             MOV 0,0,SZR
40 00061 002400             JMP @.SRET
41 00062 000324             PCRLF
42 00063 000323             MESSAGE
43 00064 004350             MSG10               ;"NO READY UNITS"
44 00065 000377             HALT
45 00066 000763             JMP .SET+1
46 00067 000000 .SRET:      0
```

```
▲ 0019  .MAIN
01
02                  ;       ********************************************
03                  ;       ***     INTERRUPT SERVICING                **
04                  ;       ********************************************
05
06 00670 054242 INTER:  STA 3,SAV3
07 00671 006310         SAVAC
08 00672 000477         INTA 1              ;WHO DID IT???
09 00673 044424         STA 1,S0
10 00674 030277         LDA 2,CDSK
11 00675 132415         SUB# 1,2,SNR
12 00676 000422         JMP INT.1           ;DISK PACK INTERRUPT
13 00677 030023         LDA 2,C10
14 00700 132415         SUB# 1,2,SNR
15 00701 000443         JMP ITTI            ;TTI INTERRUPT
16 00702 034240         LDA 3,CNIOC         ;SOMEONE ELSE
17 00703 137000         ADD 1,3
18 00704 054401         STA 3,.+1
19 00705 000000         NIOC 0              ;DEV ADDR ADDED DYNAMICALLY
20 00706 000324         PCRLF
21 00707 006323         MESSAGE
22 00710 004172         MSG9                ;INTERRUPT FROM DEVICE
23 00711 024400         LDA 1,S0
24 00712 000322         TYPZ1
25 00713 006317         SETAC
26 00714 034242         LDA 3,SAV3
27 00715 000177         INTEN
28 00716 002000         JMP @0              ;RETURN
29 00717 000000 S0:     0
30
31                      ;DISK PACK INTERRUPT
32
33 00720 000033 INT.1:  DIAC 0,.DSKP        ;READ STATUS
34 00721 040236         STA 0,ISTAT         ;SAVE FOR ALL TO USE
35 00722 030167         LDA 2,UNIT
36 00723 010420         ISZ WTRET
37 00724 002417         JMP @WTRET
38
39                      ;WAIT FOR INTERRUPT
40                      ;RUN TIMER TO PREVENT HANGUP
41
42 00725 000001 WAIT:   LDA 0,M1SD.         ;WAIT 2 SEC.
43 00726 120400         SUB 1,1
44 00727 034414         STA 3,WTRET         ;USE # IN ACS FOR TIME
45 00730 000177         INTEN               ;ENABLE INTERRUPTS
46 00731 006313         DLY1S               ;TIME FOR 12.5MS
47 00732 101124         INC 0,0,SZR
48 00733 000770         JMP .-2
49
50 00734 000077         INTDS               ;TIMEOUT, DISABLE INTER.
51 00735 000077         IORST               ;CLEAR ADAPTER FOR 2ND COMP.
52 00736 000000         HEADER              ;PRINT
53 00737 000324         PCRLF               ;ERROR MESSAGE
54 00740 000000         MESSAGE
55 00741 004000         MSG0                ;"INTERRUPT TIMEOUT"
56 00742 002721         JMP @WTRET          ;ERROR RETURN (+1)
57 00743 000000 WTRET:  0
```

```
▲ 0020  .MAIN
01
02                              ;TELETYPE INTERRUPT
03
04 00744 060210 ITTI:   NIOC TTI          ;KNOCK DOWN THE TTI FLAG
05 00745 020156         LDA 0,CSIF        ;CSI FLAG (SA 405)
06 00746 101004         MOV 0,0,SZR
07 00747 002356         JMP @ICSI         ;RETURN TO CSI
08 00750 020270         LDA 0,EVODN       ;IF TWO COMPUTERS
09 00751 101005         MOV 0,0,SNR       ;ONLY PRINT DURING
10 00752 000404         JMP .+4           ;READ OR WRITE
11 00753 020164         LDA 0,MODE
12 00754 101225         MOVZR 0,0,SNR
13 00755 000433         JMP ITT.5         ;A SEEK OR RECAL
14 00756 006324         PCRLF
15 00757 006324         PCRLF
16 00760 020157         LDA 0,UNIT
17 00761 040433         STA 0,SAVU
18 00762 020310         LDA 0,ALLRET      ;SAVE ALL RETURN
19 00763 040432         STA 0,SAVA
20 00764 020544         LDA 0,POT-1
21 00765 040021         STA 0,IDX1
22 00766 022021 ITT.1:  LDA 0,@IDX1       ;PRINT TABLE
23 00767 101005         MOV 0,0,SNR
24 00770 000414         JMP ITT.4         ;END TABLE, DONE.
25 00771 040405         STA 0,ITT.2
26 00772 022021         LDA 0,@IDX1
27 00773 040405         STA 0,ITT.3
28 00774 006324         PCRLF
29 00775 006323         MESSAGE
30 00776 000000 ITT.2:  0                 ;TITLE
31 00777 006314         DOALL             ;PRINT FOR ALL RDY UNITS
32 01000 000000 ITT.3:  0                 ;PRINT ROUTINE ADDRESS
33 01001 063610         SKPDN TTI         ;QUIT IF KEY STRUCK
34 01002 000764         JMP ITT.1
35 01003 060210         NIOC TTI
36 01004 020410 ITT.4:  LDA 0,SAVU        ;RESTORE UNIT #
37 01005 040157         STA 0,UNIT
38 01006 020407         LDA 0,SAVA        ;RESTORE ALL RETURN
39 01007 040310         STA 0,ALLRET
40 01010 006317 ITT.5:  SETAC             ;RESTORE AC'S
41 01011 034242         LDA 3,SAV3
42 01012 063177         INTEN
43 01013 002000         JMP @0
44 01014 000000 SAVL:   0
45 01015 000000 SAVA:   0
46
47                              ;CLEAR READ BUFFER
48
49 01016 122400 CLRB:   SUB 0,0           ;CLEAR READ BUFFER
50 01017 030000         LDA 2,BUFF        ;SPACE TO ZEROS
51 01020 034002         LDA 1,SC
52 01021 125040         MOVS 1,1
53 01022 125000         ADD 2,1
54 01023 041000         STA 0,0,2
55 01024 151400         INC 2,2
56 01025 152414         SUBS 1,2,SZR
57 01026 000775         JMP .-3
58 01027 001400         JMP 0,3
```

```
A 0001   .MAIN
01
02                              ;SET PARAMETERS
03
04 01030 004403 SET:    DTA 3,SERET
05 01031 030137         LDA 2,UNIT      ;SETUP DISK CONTROL
06 01032 020002         LDA 0,DC        ;FOR CURRENT INSTRUCTION
07 01033 040300         DTA 0,.DC
00 01034 034207         LDA 3,SMAX      ;MAX # SECT DUE TO MEM SIZE
09 01035 110433         SUBZ# 0,3,SNC
10 01036 000400         JMP S,1         ;MEMORY TOO SMALL
11 01037 100400         NEG 0,0         ;CA=HEAD=SECTOR
12 01040 024201         LDA 1,C17       ;SECTOR COUNT
13 01041 107400         AND 0,1
14 01042 021020         LDA 0,UNTINS,2
15 01043 107000         ADD 0,1
16 01044 020201         LDA 0,SEC       ;FIRST SECTOR
17 01045 103120         ADDZL 0,0
18 01046 103120         ADDZL 0,0
19 01047 107000         ADD 0,1
20 01050 020203         LDA 0,HD
21 01051 101300         MOVS 0,0
22 01052 107000         ADD 0,1
23 01053 007233         DOCC 1,.DSKP    ;LOAD UNIT=HEAD=SECT=CNT
24 01054 002433         DIC 0,.DSKP     ;READ IT BACK
25 01055 122415         SUB# 1,0,SNR    ;CHECK
26 01056 000400         JMP SET1
27 01057 000310         SAVAC           ;LOAD ERROR
28 01060 000324         PCRLF
29 01061 000325         MESSAGE
30 01062 004472         MSGS3           ;"DOC"
31 01063 000412         JMP SET2
32
33 01064 024254 SET1:   LDA 1,CA
34 01065 000033         DOB 1,.DSKP     ;LOAD CURRENT ADDRESS
35 01066 001433         DIB 0,.DSKP     ;READ IT BACK
36 01067 100410         SUB# 0,1,SNR    ;CHECK
37 01070 002420         JMP SERET
38 01071 000310         SAVAC           ;LOAD ERROR
39 01072 000324         PCRLF
40 01073 000325         MESSAGE
41 01074 004475         MSG4            ;"DOD"
42 01075 000000 SET2:   MESSAGE
43 01076 004333         MSG5            ;"LOAD ERROR"
44 01077 000324         PCRLF
45 01100 000000         MESSAGE
46 01101 004402         MSG1            ;GOOD
47 01102 024243         LDA 1,SAV1
48 01103 000001         TYPAC1
49 01104 000325         MESSAGE
50 01105 004407         MSG2            ;BAD
```

```
A 0022  .MAIN
01
02 01106 024237          LDA 1,SAV0
03 01107 000321          TYPAC1
04 01110 030167          LDA 2,UNIT        ;COUNT AS A MISC ERROR
05 01111 011151          ISZ MISC,2
06 01112 000354          CKSW              ;CHECK SWITCHES
07 01113 000710          JMP SET+1
08 01114 002401          JMP @SERET
09 01115 000000 SERET:   0
10
11 01116 006316 S.1:     SAVAC
12 01117 006324          PCRLF
13 01120 006323          MESSAGE
14 01121 004422          MSG39             ;"MEM TOO SMALL
15 01122 024237          LDA 1,SAV0
16 01123 006322          TYPZ1
17 01124 006323          MESSAGE
18 01125 004435          MSG40             ;SECTORS. RESTART
19 01126 063077          HALT
20 01127 000400          JMP .
21
22                       ;PRINT TABLE
23
24 01130 001130          .
25 01131 004206 POT:     MSG13
26 01132 001160          PHDR
27 01133 004221          MSG15
28 01134 001170          PTSEK
29 01135 004212          MSG14
30 01136 001172          PSKER
31 01137 004314          MSG25
32 01140 001201          PTWW
33 01141 004324          MSG26
34 01142 001200          PTWR
35 01143 004361          MSG29
36 01144 001215          PTAER
37 01145 004371          MSG30
38 01146 001217          PTPAE
39 01147 004333          MSG27
40 01150 001211          PTCWE
41 01151 004345          MSG28
42 01152 001213          PTCWP
43 01153 004434          MSG31
44 01154 001221          PTDER
45 01155 004351          MSG32
46 01156 001223          PTPCE
47 01157 004357          MSG33
48 01160 001227          PTCRD
49 01161 004345          MSG34
50 01162 001231          PTPCW
51 01163 004343          MSG35
52 01164 001235          PTMS
53 01165 000000          0
```

```
A 0023  .MAIN.
01
02 01100 054440 PHDR:  STA 3,TRET       ;PRINT HEADER NUMBER
03 01101 024107        LDA 1,UNIT
04 01170 000322        TYPZ1
05 01171 002442        JMP @TRET
06
07 01172 025065 PSKER: LDA 1,SEKER,2    ;PRINT # SEEK ERRORS
08 01173 054440 PS.1:  STA 3,TRET
09 01174 000320        TYPDEC
10 01175 002430        JMP @TRET
11
12 01176 025055 PTSEK: LDA 1,SEEKT,2    ;PRINT # SEEKS
13 01177 031061        LDA 2,SEEKT+4,2
14 01200 000403        JMP PT.1
15
16 01201 025071 PTWW:  LDA 1,WDSW,2     ;PRINT WORDS WRITTEN
17 01202 031075        LDA 2,WDSW+4,2
18 01203 054430 PT.1:  STA 3,TRET
19 01204 000342        JSR @I.DBD
20 01205 002420        JMP @TRET
21
22 01206 025101 PTWR:  LDA 1,WDSR,2     ;PRINT WORDS READ
23 01207 031105        LDA 2,WDSR+4,2
24 01210 000773        JMP PT.1
25
26 01211 025111 PTCWE: LDA 1,CWER,2     ;PRINT CHECK WORD ERRORS
27 01212 000761        JMP PS.1
28
29 01213 025115 PTCWP: LDA 1,CWERP,2    ;PRINT PERM CHK WD ERRS
30 01214 000757        JMP PS.1
31
32 01215 025121 PTAER: LDA 1,ADDER,2    ;PRINT ADDRESS ERRORS
33 01216 000755        JMP PS.1
34
35 01217 025125 PTPAE: LDA 1,PADER,2    ;PRINT PERM ADDR ERRS
36 01220 000753        JMP PS.1
37
38 01221 025131 PTDER: LDA 1,DATER,2    ;PRINT DATA ERRORS
39 01222 000751        JMP PS.1
40
41 01223 025145 PTPDE: LDA 1,PDER,2     ;PRINT PERM DATA ERRS
42 01224 000747        JMP PS.1
43
44 01225 025151 PTMS:  LDA 1,MISC,2     ;PRINT # MISC ERRORS
45 01226 000745        JMP PS.1
46
47 01227 025135 PTCWD: LDA 1,CWDE,2     ;PRINT # CHK WD AND
48 01230 000743        JMP PS.1         ;DATA ERRORS.
49
50 01231 025141 PTPCW: LDA 1,CWDEP,2    ;PRINT # PERM CHECK WORD
51 01232 000741        JMP PS.1         ;AND DATA ERRORS
52
53 01233 000000 TRET:  0
```

```
▲ 0024   .MAIN
01
02                          ;EXECUTE THE ROUTINE POINTED TO BY'
03                          ;THE WORD FOLLOWING THE CALL.  DO TO
04                          ;ONCE FOR EACH ACTIVE UNIT
05
06 01234 054310 ALL:        STA 3,ALLRET
07 01235 102400             SUB 0,0
08 01236 040157 ALL.1:      STA 0,UNIT
09 01237 034310             LDA 3,ALLRET
10 01240 030157             LDA 2,UNIT
11 01241 021022             LDA 0,DRVS,2
12 01242 024160             LDA 1,RDYUNT
13 01243 123414             AND# 1,0,SZR
14 01244 007400             JSR @0,3          ;DO IT FOR THIS UNIT
15 01245 000401             JMP .+1           ;IGNORE RETURN+1,+2 ROUT'S
16 01246 000401             JMP .+1
17 01247 010157             ISZ UNIT
18 01250 020157             LDA 0,UNIT
19 01251 024024             LDA 1,C4
20 01252 122414             SUB# 1,0,SZR
21 01253 000763             JMP ALL.1
22 01254 010310             ISZ ALLRET
23 01255 002310             JMP @ALLRET
24
25
26                          ;DELAY 12.5MS
27                          ;DO NOT USE AC0,1
28
29 01256 054400 DLY:        STA 3,DLYRET
30 01257 030234             LDA 2,TIME
31 01260 050235             STA 2,TIMER
32 01261 014235             DSZ TIMER
33 01262 000777             JMP .-1
34 01263 002401             JMP @DLYRET
35 01264 000000 DLYRET:     0
36
37
38
39                          ;PRINT STATUS
40
41 01265 006333 PS:         HEADER
42 01266 000324             PCRLF
43 01267 024236             LDA 1,ISTAT
44 01270 000321             TYPAC1
45 01271 000333             MESSAGE
46 01272 034164             NOC1              ;" = ENDING STATUS"
47 01273 002274             JMP @RARET
```

```
▲ 0025  .MAIN
01
02                          ;ASK FOR DISK SIZE
03
04 01274 034420 .GSD1:   STA 3,FDSRET
05 01275 000324          PCRLP
06 01276 000323          MESSAGE
07 01277 004047          MSGGO              ;"TYPE THE NUMBER OF DISK SURFACES "
08 01300 000330          INPUT              ;WAIT FOR INPUT
09 01301 000257          UDP                ;BYTE POINTER
10 01302 003440          JMP .GSD1          ;"CR" ONLY IS ERROR
11 01303 000326          GETATM             ;GET THE NUMBER
12 01304 105000          MOV 0,1
13 01305 000325          SEARCH             ;IS NUMBER 2,10 OR 20
14 01306 001322          HNUM               ;TABLE ADDRESS
15 01307 000433          JMP .GSD1          ;NO !!
16 01310 021007 FDS,2:   LDA 0,SMX-HNUM,2
17 01311 040171          STA 0,LS           ;LAST SECTOR+1
18 01312 021004          LDA 0,HMX-HNUM,2
19 01313 040170          STA 0,LHD          ;LAST HEAD+1
20 01314 021012          LDA 0,MS1-HNUM,2;HEAD MASK
21 01315 040303          STA 0,HMSK
22 01316 021015          LDA 0,MS2-HNUM,2;SECTOR MASK
23 01317 040304          STA 0,SMSK
24 01320 002401          JMP @FDSRET
25 01321 000000 FDSRET:  0
```

```
^ 0026  .MAIN
01
02 01322 000002 HNUM:  2                ;CARTRIDGE
03 01323 000010         10              ;2311
04 01324 000020         20              ;2314
05 01325 000000         0
06
07 01326 000002 HMX:    2               ;CARTRIDGE
08 01327 000012         10.             ;2311
09 01330 000024         20.             ;2314
10
11 01331 000014 SMX:    12.
12 01332 000006         6
13 01333 000014         12.
14
15 01334 000400 MS1:    400             ;HEAD MASKS
16 01335 007400         7400
17 01336 017400         17400
18
19 01337 007400 MS2:    7400            ;SECTOR MASKS
20 01340 003400         3400
21 01341 007400         7400
22
23 01342 006323 .GSD1:  MESSAGE         ;ERROR
24 01343 004444         MSG43           ; "-?-"
25 01344 000731         JMP .GSD+1
```

```
A 0027  .MAIN
01
02                ;       ********************************************************
03                ;       ***     DISK RELIABILITY TEST                   *******
04                ;       ********************************************************
05
06 01345 102021 RELOD:  SUBZR 0,0,SKP   ;USE ODD CYLINDERS
07 01346 102020 RELEV:  SUBZL 0,0       ;USE EVEN CYLINDERS
08 01347 101001         MOV 0,0,SKP
09 01350 102400 RELALL: SUB 0,0         ;USE ALL CYLINDERS
10 01351 040070         STA 0,EVODN
11 01352 020061         LDA 0,IRAN      ;SET FOR RANDOM DATA
12 01353 042343         STA 0,@IRDAT
13 01354 042344         STA 0,@IWDAT
14 01355 020065         LDA 0,BUFF      ;MEM ADDR, CONST
15 01356 040254         STA 0,CA
16
17 01357 006361 REL.1:  RAND            ;GET RANDOM UNIT #
18 01360 030302         LDA 2,UMSK      ;UNIT SIZE MASK
19 01361 113700         ANDS 0,2
20 01362 021022         LDA 0,DRVS,2
21 01363 024160         LDA 1,RDYUNT    ;UNIT READY ?
22 01364 123403         AND 1,0,SNR
23 01365 000772         JMP REL.1       ;NO, TRY AGAIN
24 01366 050157         STA 2,UNIT
25
26 01367 006361 REL.2:  RAND            ;GET RANDOM HEAD #
27 01370 024303         LDA 1,HMSK      ;HEAD SIZE MASK
28 01371 123700         ANDS 1,0
29 01372 024170         LDA 1,LHD       ;HEAD # OK ?
30 01373 100437         SUBZ# 0,1,SBN
31 01374 000773         JMP REL.2       ;TOO LARGE, TRY AGAIN
32 01375 040263         STA 0,HD
33
34 01376 006361 REL.3:  RAND            ;GET RANDOM STARTING SECTOR
35 01377 024304         LDA 1,SMSK      ;SECTOR SIZE MASK
36 01400 123700         ANDS 1,0
37 01401 024171         LDA 1,LS        ;SECT # OK ?
38 01402 100437         SUBZ# 0,1,SBN
39 01403 000764         JMP REL.2       ;TOO LARGE, TRY AGAIN
40 01404 040261         STA 0,SEC
41
42 01405 006361 REL.4:  RAND            ;GET RANDOM # OF SEC
43 01406 024305         LDA 1,LMSK      ;# SECTORS MASK
44 01407 123703         ANDS 1,0,SNR    ;DON'T ALLOW 0
45 01410 000000         LDA 0,@0        ;ZERO IS ACTUALLY 20
46 01411 021007         LDA 1,SMAX      ;ENOUGH CORE ?
47 01412 100400         SUBZ# 0,1,SNC
48 01413 000772         JMP REL.4       ;NO, TRY AGAIN
49 01414 040262         STA 0,CC
50 01415 024261         LDA 1,SEC       ;CHECK TO SEE IF
51 01416 030263         LDA 2,HD        ;TRANSFER WILL CAUSE
52 01417 000010         SAVAC           ;"END CYLINDER"
53 01420 010267         ISZ SAVC
```

```
▲ 0028  .MAIN
01
02 01421 014237 REL.5:  DSZ SAV0
03 01422 000402          JMP .+2
04 01423 000420          JMP REL.8          ;SECTOR COUNT OK;
05 01424 010240          ISZ SAV1           ;CURRENT SECT+1
06 01425 020240          LDA 0,SAV1
07 01426 024171          LDA 1,LS           ;HEAD OVERFLOW ?
08 01427 122434          SUBZ# 1,0,SZR
09 01430 000771          JMP REL.5          ;NO, COUNT ANOTHER SECT
10 01431 102400          SUB 0,0
11 01432 040240          STA 0,SAV1
12 01433 010241          ISZ SAV2           ;YES, SELECT NEXT HEAD
13 01434 020241          LDA 0,SAV2
14 01435 024170          LDA 1,LHD          ;END CYLINDER ?
15 01436 122414          SUB# 1,0,SZR
16 01437 000762          JMP REL.5          ;NO, COUNT ANOTHER SECTOR
17 01440 020237          LDA 0,SAV0         ;YES, DONE IF THIS IS
18 01441 101224          MOVZR 0,0,SZR      ;THE LAST SECTOR
19 01442 000743          JMP REL.4          ; GET ANOTHER RAND #
20
21 01443 006361 REL.6:  RAND               ;GET RANDOM CYLINDER #
22 01444 024216          LDA 1,C377
23 01445 123400          AND 1,0
24 01446 024270          LDA 1,EVODN
25 01447 125005          MOV 1,1,SNR
26 01450 000404          JMP REL.7          ;USE ALL CYLINDERS
27 01451 101220          MOVZR 0,0          ;CYL # RIGHT ONE
28 01452 125100          MOVL 1,1           ;ODD BIT TO (C)
29 01453 101100          MOVL 0,0           ;CYL #, ODD OR EVEN
30 01454 024213 REL.7:  LDA 1,C203.        ;IS CYL # < 203.
31 01455 122432          SUBZ# 1,0,SZC
32 01456 000765          JMP REL.6          ;NO, TRY ANOTHER #
33 01457 030157          LDA 2,UNIT
34 01460 025051          LDA 1,CYLT,2
35 01461 045045          STA 1,CYLF,2       ;"TO"="FROM"
36 01462 041051          STA 0,CYLT,2       ;NEW "TO"
37
38                       ;**
39                       ;**      SEEK
40                       ;**
41
42 01463 102000          ADC 0,0
43 01464 040155          STA 0,ASW
44 01465 006330          SEEK               ;SEEK NEW CYLINDER
45 01466 006334          CASW               ;ERROR, CHECK SWITCHES
46 01467 000776          JMP .-1            ;LOOP ON IT
```

```
▲ 0029  .MAIN
01
02                          ;**
03                          ;**      WRITE DISK
04                          ;**
05
06 01470 102000             ADC 0,0
07 01471 040155             STA 0,HSW
08 01472 020243             LDA 0,RANDOM     ;SAVE BEGINNING OF
09 01473 040245             STA 0,RELRAN     ;RANDOM NUMBERS
10 01474 020245             LDA 0,RELRAN     ;RESET RANDOM NUMBERS
11 01475 040243             STA 0,RANDOM     ;FOR LOOPING
12 01476 006347             WRITE            ;GEN DATA & WRITE
13 01477 006354             CKSW             ;ERROR, CHECK SWITCHES
14 01500 000774             JMP .-4          ;LOOP ON IT
15
16 01501 020270             LDA 0,EVODN      ;IF TWO COMPUTERS, SEEK AGAIN
17 01502 101005             MOV 0,0,SNR
18 01503 000406             JMP .RED         ;ONLY ONE
19 01504 102000             ADC 0,0
20 01505 040155             STA 0,HSW
21 01506 006350             SEEK             ;SEEK SAME CYLINDER
22 01507 006354             CKSW             ;ERROR, CHECK SWITCH CONTROL
23 01510 000776             JMP .-2          ;LOOP ON IT
24
25                          ;**
26                          ;**      READ DISK
27                          ;**
28
29 01511 102000 .RED:       ADC 0,0
30 01512 040155             STA 0,HSW
31 01513 020245             LDA 0,RELRAN     ;SET RAN # GEN FOR THE
32 01514 040243             STA 0,RANDOM     ;READ CHECK
33 01515 006346             READ             ;READ AND CHECK DATA
34 01516 006354             CKSW             ;ERROR, CHECK SWITCHES
35 01517 000774             JMP .-4          ;LOOP ON IT
36
37                          ;CHECK ENDING FOR TERMINATION
38
39 01520 020161             LDA 0,RALL
40 01521 101005             MOV 0,0,SNR
41 01522 000035             JMP REL.1        ;MORE RELIABILITY
42 01523 014402             DSZ CRALL        ;RUN ALL MODE
43 01524 000000             JMP REL.1        ;DO 512 LOOPS ONLY
44 01525 006324             PCRLF
45 01526 006333             MESSAGE
46 01527 034441             MSG41            ;"PASS"
47
48
49
50                          .EOT
```

```
  0030  .MAIN
01                ;       ************************************************
02                ;       *******          RUN ALL TESTS          *******
03                ;       ************************************************
04
05                        ;RUN ALL DISK ADDRESS PATTERNS
06                        ;THEN RUN RANDOM EXERCISER FOR A
07                        ;WHILE.  THEN REPEAT
08
09 01530 020173 RUNALL: LDA 0,C1000
10 01531 040414         STA 0,CRALL
11 01532 040161         STA 0,RALL
12 01533 030414         LDA 2,A1-1
13 01534 050412         STA 2,PADD
14 01535 010411 R.1:    ISZ PADD
15 01536 022410         LDA 0,@PADD
16 01537 101005         MOV 0,0,SNR
17 01540 000610         JMP RELALL
18 01541 040275         STA 0,DADAT
19 01542 006314         DOALL                   ;DO DISK ADDRESS
20 01543 001615         DAT.
21 01544 000771         JMP R.1                 ;TEST FOR EACH PATTERN
22 01545 000000 CRALL:  0
23 01546 000000 PADD:   0
24
25 01547 001547         .
26 01550 004134 A1:     ZEROS
27 01551 004135         ONES
28 01552 004140         PAT1
29 01553 004142         FL1
30 01554 004147         FLZ
31 01555 003377         RAN
32 01556 000000         0
```

```
A 0031   .MAIN
01
02 01557 020770 DATR:   LDA 0,A1+3       ;DISK ADDR TEST
03 01560 024244         LDA 1,.RAN       ;SET RANDOM BASE
04 01561 044243         STA 1,RANDOM     ; NUMBER
05 01562 000412         JMP DATF0+1
06
07 01563 020700 DAT1:   LDA 0,A1+1
00 01564 000410         JMP DATF0+1
09
10 01565 020703 DAT0:   LDA 0,A1
11 01566 000406         JMP DATF0+1
12
13 01567 020703 DATP:   LDA 0,A1+2
14 01570 000404         JMP DATF0+1
15
16 01571 020762 DATF1:  LDA 0,A1+3
17 01572 000402         JMP DATF0+1
10
19 01573 020701 DATF0:  LDA 0,A1+4
20 01574 040275         STA 0,DADAT
21 01575 000210         NIOC TTI
22 01576 006324         PCRLF
23 01577 006323         MESSAGE
24 01600 004700         MSG71            ;"SET SW4 FOR INTERCHANGE....
25 01601 006324         PCRLF
26 01602 006323         MESSAGE
27 01603 004720         MSG72            ;"STRIKE ANY KEY TO CONTINUE"
28 01604 003610         SKPDN TTI
29 01605 000777         JMP .-1
30 01606 000210         NIOC TTI
31 01607 006314 DAT:    DOALL            ;DO IT ONCE FOR EACH
32 01610 001615         DAT.             ;READY DISK
33 01611 006324         PCRLF
34 01612 006323         MESSAGE
35 01613 004441         MSG41            ;"PASS"
36 01614 000773         JMP DAT
```

```
▲ 0032  .MAIN
01
02                ;       ********************************************
03                ;       *******          DISK ADDRESS TEST       *******
04                ;       ********************************************
05
06                ;       WRITE THE ENTIRE DISK, THEN READ.
07                ;       DATA PATTERN IN (DADAT).
08                ;       DISK DRIVE IN (UNIT).
09
10 01615 054525 DAT.1:   STA 3,DARET
11 01616 074477          READS 3              ;IF SW5=1, READ ONLY
12 01617 024172          LDA 1,C2000
13 01620 020524          LDA 0,DRD
14 01621 137404          AND 1,3,SZR
15 01622 000403          JMP .+3
16 01623 126000          ADC 1,1
17 01624 020521          LDA 0,DWT
18 01625 044516          STA 1,RFLG           ;R/W FLAG
19 01626 040454          STA 0,DATRW          ;WRITE FIRST
20 01627 020247          LDA 0,LAST
21 01630 040254          STA 0,CA             ;BUFFER ADDRESS
22 01631 020275          LDA 0,DADAT
23 01632 042343          STA 0,@IRDAT         ;DEFINE DATA
24 01633 042344          STA 0,@IWDAT         ;PATTERN
25 01634 020243          LDA 0,RANDOM
26 01635 040511          STA 0,DARAN          ;SAVE RANDOM START
27
28 01636 020510 DAT.0:   LDA 0,DARAN          ;RESET RANDOM FOR
29 01637 040243          STA 0,RANDOM         ;RECYCLE
30 01640 102400          SUB 0,0
31 01641 040036          STA 0,FLO1           ;RESET FLOATING 0&1
32 01642 040037          STA 0,FLO2           ;PATTERNS
33 01643 102000          ADC 0,0
34 01644 040155          STA 0,HSW            ;SET HEADER SWITCH
35 01645 006351          RECAL                ;RECALIBRATE DRIVE
36 01646 006354          CKSW                 ;ERROR, CHECK SWITCHES
37 01647 000776          JMP .-2              ;LOOP ON ERROR
38
39 01650 030157 DAT.1:   LDA 2,UNIT           ;SEEK NEW CYLINDER
40 01651 021051          LDA 0,CYLT,2         ;TO = FROM
41 01652 041045          STA 0,CYLF,2
42 01653 011051          ISZ CYLT,2           ;TO+1 = TO
43 01654 101400          INC 0,0
44 01655 024213          LDA 1,C203.
45 01656 122403          SUB 1,0,SNR
46 01657 000437          JMP DAT.3            ;DONE, READ OR WRITE
47 01660 102000          ADC 0,0              ;SET HEADER SWITCH
48 01661 040155          STA 0,HSW
49 01662 006352          SEEK
50 01663 006354          CKSW                 ;ERROR, CHECK SWITCHES
51 01664 000776          JMP .-2              ;LOOP ON ERROR
52 01665 102400          SUB 0,0              ;BEGIN WITH HEAD 0
53 01666 040155          STA 0,HD             ;SECTOR 0
54 01667 040154          STA 0,SEC
55 01670 040275          STA 0,CYLC
56 01671 024207          LDA 1,SMAX           ;MAX # OF SECTORS
57 01672 042252          STA 0,SC             ;DUE TO MEM SIZE
58 01673 040303          STA 0,CSC
```

```
A 0033  .MAIN
01
02 01674 102000 DAT.2:  ADC 0,0          ;SET THE HEADER SWITCH
03 01675 040103         STA 0,HSW
04 01676 020243         LDA 0,RANDOM     ;SAVE RAN # POSITION
05 01677 040245         STA 0,RELRAN     ;IN CASE OF SCOPE LOOP
06 01700 020245         LDA 0,RELRAN     ;RESTORE RANDOM #
07 01701 040243         STA 0,RANDOM     ;FOR SCOPE LOOP
08 01702 000347 DATRW:  WRITE            ;READ OR WRITE
09 01703 000354         CKSW             ;ERROR, CHECK SWITCHES
10 01704 000774         JMP .-4          ;LOOP ON ERROR
11 01705 020270         LDA 0,CFLG       ;END CYL NOW ?
12 01706 101004         MOV 0,0,SZR
13 01707 000741         JMP DAT.1        ;YES - SEEK AGAIN
14 01710 004437         JSR SCNT         ;ADVANCE DISK ADDRESS
15 01711 040251         STA 0,SEC        ;UPDATE SECTOR AND
16 01712 044253         STA 1,HD         ;HEAD.
17 01713 004434         JSR SCNT         ;LOOK AHEAD
18 01714 050252         STA 2,SC         ;SET NEW SECTOR COUNT
19 01715 000757         JMP DAT.2
20
21 01716 010425 DAT.3:  ISZ RFLG
22 01717 000404         JMP DAT.4        ;DONE WRITE & READ
23 01720 020424         LDA 0,DRD
24 01721 040701         STA 0,DATRW      ;SET TO READ MODE
25 01722 000714         JMP DAT.0
26 01723 000477 DAT.4:  READS 0          ;IF SW4=1 BREAD FOR
27 01724 024035         LDA 1,C4K        ;DISK INTERCHANGE
28 01725 107404         AND 0,1,SZR
29 01726 000407         JMP DAT.5
30 01727 024172         LDA 1,C2000      ;IF SW5=1 RESET
31 01730 107403         AND 0,1,SNR      ;RANDOM PATTERN
32 01731 002411         JMP @DARET       ;EXIT
33 01732 024244         LDA 1,.RAN       ;RANDOM INITIAL
34 01733 044243         STA 1,RANDOM     ;START
35 01734 002400         JMP @DARET
36 01735 000324 DAT.5:  PCRLF
37 01736 000323         MESSAGE
38 01737 004030         MSGGB            ;"INTERCHANGE DISK"
39 01740 003077         HALT
40 01741 000777         JMP .-1
41 01742 000000 DARET:  0
42 01743 000000 RFLG:   0
43 01744 000345 DRD:    READ
44 01745 000347 DWT:    WRITE
45 01746 000038 DATAA:  38
```

```
A 0034   .MAIN
01
02                ;       MOVE BEGINNING SECTOR AND HEAD ADDRESS
03                ;       FORWARD BY THE NUMBER OF SECTORS IN (CSC).
04                ;       IF END CYLINDER SET "CFLG".
05
06                ;       EXIT =  AC0= NEW SECTOR START
07                ;               AC1= NEW HEAD START
08                ;               AC2= SECTOR COUNT
09
10 01747 054430 SCNT:   STA 3,SCRET
11 01750 020300         LDA 0,CSC
12 01751 040427         STA 0,WORK
13 01752 020251         LDA 0,SEC       ;CURRENT BEGINNING SECTOR
14 01753 024253         LDA 1,HD        ;CURRENT BEGINNING HEAD
15
16 01754 101400 SC.1:   INC 0,0         ;SECT+1
17 01755 030171         LDA 2,LS        ;CHECK FOR SECTOR OVERFLOW
18 01756 112405         SUB 0,2,SNR
19 01757 000405         JMP SC.3        ;OVERFLOW
20 01760 014420 SC.2:   DSZ WORK        ;DONE ?
21 01761 000773         JMP SC.1        ;NO
22 01762 030300         LDA 2,CSC       ;YES, EXIT
23 01763 002414         JMP @SCRET
24
25 01764 141000 SC.3:   MOV 2,0         ;SECT=0
26 01765 125400         INC 1,1         ;HEAD+1
27 01766 030170         LDA 2,LHD
28 01767 132404         SUB 1,2,SZR     ;HEAD OVERFLOW ?
29 01770 000770         JMP SC.2        ;NO
30 01771 010275         ISZ CFLG        ;YES, END CYLINDER
31 01772 034400         LDA 3,WORK
32 01773 030300         LDA 2,CSC
33 01774 172400         SUB 3,2
34 01775 151400         INC 2,2
35 01776 002401         JMP @SCRET
36 01777 000000 SCRET:  0
37 02000 000000 WORK:   0
```

```
                        ; ******************************************************
                        ; ******       COMMAND STRING INTERPRETER        *****
                        ; ******************************************************

                        ;GET COMMAND STRING PARAMETERS
                        ;UNIT SELECT
02001 006323 QUEST:     MESSAGE
02002 004444            MSG43           ; -?-
02003 102520 CMDST:     SUBZL 0,0       ;SET THE CSI FLAG
02004 040150            STA 0,CSIF
02005 060210            NIOC TTI        ;CLEAR TTI FLAG
02006 060233            NIOC .DSKP      ;CLEAR DISK GARBAGE
02007 006324            PCRLF
02010 006323            MESSAGE
02011 004240            MSG17           ;"UNIT: "
02012 006330            INPUT           ;ACCEPT INPUT
02013 000257            USP
02014 000417            JMP CMD.2       ;SAME AS LAST TIME
02015 102400            SUB 0,0
02016 040160            STA 0,RDYUNT    ;CLEAR ALL UNITS
02017 006326            GETATM
02020 125004            MOV 1,1,SZR     ;AC0=# AC1=NAME
02021 000760            JMP QUEST       ;NAMES NOT LEGAL
02022 024024            LDA 1,C4
02023 122432            SUBZ# 1,0,SZC
02024 000755            JMP QUEST       ;UNIT #>3
02025 040137            STA 0,UNIT
02026 101003            MOV 0,0,SNC
02027 000752            JMP QUEST       ;MORE LEFT, ERROR
02030 000403            JMP CMD.2

                        ;DATA SELECT

02031 006323            MESSAGE
02032 004444            MSG43           ; -?-
02033 006324 CMD.2:     PCRLF
02034 006323            MESSAGE
02035 004244            MSG16           ;"DATA: "
02036 006330            INPUT           ;ACCEPT INPUT
02037 000257            USP
02040 [illegible]       JMP GCS         ;SAME AS LAST TIME
02041 [illegible]       GETATM
02042 125005            MOV 1,1,SNR
[illegible] [illegible] JMP CMD.6       ;NO NAME, MUST BE #
[illegible] [illegible] SEARCH          ;SEARCH NAME TABLE
[illegible] [illegible] DNT
02046 000763            JMP CMD.2-2     ;NOT FOUND
02047 [illegible]       LDA 1,DNT.1-DNT,2
02050 [illegible]       STA 1,PIRDAT
02051 [illegible]       STA 1,PIWDAT
02052 [illegible]       MOV 2,0,SNR
[illegible] [illegible] JMP CMD.2-2     ;NO OR FOLLOWING NAME
[illegible] [illegible] JMP GCS         ;GET COMMAND STRING
```

```
▲ 0036  .MAIN
01
02 02055 024261 CMD.3:  LDA 1,VARST
03 02056 044020          STA 1,IDX0
04 02057 042020          STA 0,@IDX0
05 02060 024253          LDA 1,TERM
06 02061 125004          MOV 1,1,SZR
07 02062 000416          JMP CMD.5
08 02063 006326 CMD.4:   GETATM
09 02064 125004          MOV 1,1,SZR
10 02065 000744          JMP CMD.2-2      ;NAMES NOT ALLOWED
11 02066 042020          STA 0,@IDX0
12 02067 101002          MOV 0,0,SZC
13 02070 000410          JMP CMD.5
14 02071 020262          LDA 0,VARED
15 02072 024020          LDA 1,IDX0
16 02073 122414          SUB# 1,0,SZR
17 02074 000767          JMP CMD.4
18 02075 006324          PCRLF
19 02076 006323          MESSAGE
20 02077 004261          MSG20
21
22 02100 020020 CMD.5:   LDA 0,IDX0
23 02101 040263          STA 0,VARSP
24 02102 020333          LDA 0,IVAR
25 02103 042343          STA 0,@IRDAT
26 02104 042344          STA 0,@IWDAT
27
28
29
30                       ;GET COMMAND STRING
31
32 02105 006324 GCS:     PCRLF
33 02106 006323          MESSAGE
34 02107 004250          MSG19            ;"COMMAND STRING: "
35 02110 006330          INPUT            ;ACCEPT INPUT
36 02111 000200          CSBP             ;CMD STR BYTE POINTER
37 02112 000401          JMP .+1          ;SAME AS BEFORE
38 02113 063710 GCS.1:   SKPDZ TTI
39 02114 000667          JMP CMDST        ;INTERRUPT CMD STRING
40 02115 102000          ADC 0,0
41 02116 040135          STA 0,HSW
42 02117 006326          GETATM           ;LINCT=BYTE POINTER
43 02120 006325          SEARCH
44 02121 004102          DCT              ;NAME IN AC1
45 02122 000402          JMP CSER         ;NOT FOUND
46 02123 000207          JMP @DCT,1-DCT,0;CIRDAT+
47
48 02124 006323 CSER:    MESSAGE          ;COMMAND STRING ERROR
49 02125 004444          MSG45            ; -?-
50 02126 000767          JMP GCS
```

```
A 0037   .MAIN
01
02                          ;READ COMMAND
03
04 02127 101002 RE1:        MOV 0,0,SZC
05 02130 000774             JMP CSER            ;CR TERMINATOR ILLEGAL
06 02131 000327             GETPAR              ;GET R/W PARAMETERS
07 02132 020245             LDA 0,RELRAN        ;GET RAN # BEGIN
08 02133 040243             STA 0,RANDOM
09 02134 000346             READ                ;READ AND CHECK DATA
10 02135 000354             CKSW                ;ERROR, LOOK AT SWITCHES
11 02136 000774             JMP .-4             ;LOOP ON ERROR
12 02137 020200 RE1:        LDA 0,TERM
13 02140 101004             MOV 0,0,SZR
14 02141 000642             JMP CMDST                   ;ALL DONE
15 02142 000751             JMP GCS.1           ;GET ANOTHER COMMAND
16
17                          ;WRITE COMMAND
18 02143 101002 WT:         MOV 0,0,SZC
19 02144 000700             JMP CSER            ;CR TERMINATOR ILLEGAL
20 02145 000327             GETPAR              ;GET R/W PARAMETERS
21 02146 020243             LDA 0,RANDOM        ;SAVE BEGIN OF RANDOM
22 02147 040245             STA 0,RELRAN        ;NUMBERS IN CASE OF LOOP
23 02150 020243             LDA 0,RELRAN        ;RESET RAN # BEGINNING IN
24 02151 040243             STA 0,RANDOM        ;THE SCOPE LOOP
25 02152 000347             WRITE               ;GENERATE DATA AND WRITE
26 02153 000354             CKSW                ;ERROR, CHECK SWITCHES
27 02154 000774             JMP .-4             ;LOOP ON ERROR
28 02155 000762             JMP RE1
```

```
A 0038  .MAIN
01
02                              ;SEEK COMMAND
03
04 02156 101002 SK:     MOV 0,0,SZC
05 02157 000745         JMP CSER        ;CR TERMINATOR ILLEGAL
06 02160 006326         GETATM          ;GET CYL # IN AC0
07 02161 152560         SUBCL 2,2
08 02162 050256         STA 2,TERM
09 02163 125004         MOV 1,1,SZR
10 02164 000740         JMP CSER        ;NAMES ILLEGAL
11 02165 030157         LDA 2,UNIT
12 02166 025051         LDA 1,CYLT,2    ;TO = CYL #
13 02167 045045         STA 1,CYLF,2    ;TO BECOMES FROM
14 02170 041051         STA 0,CYLT,2    ;SET NEW TO
15 02171 006350         SEEK            ;DO THE SEEK
16 02172 006354         CKSW            ;ERROR, LOOK AT SWITCHES
17 02173 000776         JMP .-2         ;LOOP ON ERROR
18 02174 000743         JMP RE1
19
20                              ;RECALIBRATE
21
22 02175 102560 RCL:    SUBCL 0,0
23 02176 040256         STA 0,TERM      ;SAVE (C), 1=TERM
24 02177 006351         RECAL           ;RECALIBRATE
25 02200 006354         CKSW            ;ERROR, CHECK SWITCHES
26 02201 000776         JMP .-2         ;LOOP ON ERROR
27 02202 000735         JMP RE1
28
29                              ;LOOP
30
31 02203 020260 LUP:    LDA 0,CSBP      ;RESET THE COMMAND STRING
32 02204 040250         STA 0,LINCT     ;BYTE POINTER
33 02205 102000         ADC 0,0         ;SET THE HEADER FLAG
34 02206 040155         STA 0,HSW
35 02207 000704         JMP GCS.1
36
37                              ;DELAY
38
39 02210 101002 DLAY:   MOV 0,0,SZC
40 02211 000713         JMP CSER
41 02212 006326         GETATM          ;GET DELAY #
42 02213 125005         MOV 1,1,SNR
43 02214 101002         MOV 0,0,SZC     ;NAMES ILLEGAL
44 02215 000707         JMP CSER        ;CR TERM IS ILLEGAL
45 02216 123401         NEG 0,0
46 02217 006315         DLY:2           ;12.5MS DELAY LOOP
47 02220 101404         INC 0,0,SZR
48 02221 000776         JMP .-2
49 02222 000671         JMP GCS.1
50
```

```
▲ 0039  .MAIN
01
02                ;       ********************************************************
03                ;       *****           DISK HANDLER SUBROUTINES          *****
04                ;       ********************************************************
05
06                        ;RECALIBRATE
07                        ;RETURN+1  FATAL ERROR
00                        ;RETURN+3  NORMAL
09
10 02223 054274 RECL:     STA 3,RWRET
11 02224 102400           SUB 0,0             ;SET THE PROGRAM MODE
12 02225 040164           STA 0,MODE
13 02226 030157           LDA 2,UNIT
14 02227 021020           LDA 0,UNTINS,2
15 02230 063233           DOCC 0,.DSKP        ;SELECT UNIT
16 02231 020217           LDA 0,C1400
17 02232 061333           DOAP 0,.DSKP        ;RECAL
18 02233 021051           LDA 0,CYLT,2
19 02234 041045           STA 0,CYLF,2        ;TO = FROM
20 02235 102000           ADC 0,0
21 02236 041051           STA 0,CYLT,2
22 02237 020232           LDA 0,M250.         ;3 SECOND TIMER
23 02240 000341           JSR @WAT            ;WAITING FOR INTERRUPT
24 02241 002274           JMP @RWRET          ;TIMEOUT!!!
25 02242 025032           LDA 1,UNTDN,2       ;RECAL INT. RETURN
20 02243 107403           AND 0,1,SNR
27 02244 002352           PSTAT               ;NO DONE STATUS
28 02245 124000           COM 1,1
29 02246 034225           LDA 3,CSP1          ;177677
30 02247 107400           AND 3,1
31 02250 107404           AND 0,1,SZR
32 02251 002352           PSTAT               ;BAD STATUS
33 02252 000535           JMP GENRET
```

```
A 0040   .MAIN
01
02                          ;SEEK SUBROUTINE
03                          ;RETURN+3 NORMAL
04                          ;RETURN+1 FATAL ERROR
05
06 02253 054274 .SEEK:     STA 3,RWRET
07 02254 102320            SUBZL 0,0          ;SET THE PROGRAM MODE
08 02255 040164            STA 0,MODE
09 02256 030157            LDA 2,UNIT
10 02257 021026            LDA 0,UNTINS,2
11 02260 063233            DOCC 0,.DSKP       ;SELECT UNIT
12 02261 021051            LDA 0,CYLT,2       ;CYL #
13 02262 024173            LDA 1,C1000
14 02263 123000            ADD 1,0
15 02264 063333            DOAP 0,.DSKP       ;SEEK!!
16 02265 011061            ISZ SEEKT+4,2      ;COUNT SEEKS
17 02266 000402            JMP .+2
18 02267 011055            ISZ SEEKT,2        ;DOUBLE PRES.
19 02270 006360            INTWT              ;WAIT FOR INTERRUPT
20 02271 002274            JMP @RWRET         ;TIMEOUT!!!
21
22 02272 025032            LDA 1,UNTDN,2      ;INTERRUPT RETURN AC2=UNIT #
23 02273 107415            AND# 0,1,SNR       ;AC0 = STATUS
24 02274 000407            JMP SE.1           ;IMPROPER DONE STATUS
25 02275 124000            COM 1,1
26 02276 034225            LDA 3,CSP1         ;177677
27 02277 107400            AND 3,1
28 02300 107404            AND 0,1,SZR
29 02301 000402            JMP SE.1           ;WRONG STATUS
30 02302 000505            JMP GENRET
31
32 02303 024205 SE.1:      LDA 1,C40          ;SEEK ERR ??
33 02304 107404            AND 0,1,SZR
34 02305 011063            ISZ SEKER,2        ;COUNT SEEK ERRORS
35 02306 002352            PSTAT
```

```
A 0041  .MAIN
01
02                          ;WRITE SUBROUTINE
03                          ;RETURN+3, NORMAL
04                          ;RETURN+1, FATAL ERROR
05
06 02307 002326             WDATA+1         ;RETRY ADDRESS
07 02310 054274 .WRITE:     STA 3,RWRET
08 02311 020776             LDA 0,.WRITE-1  ;SET THE RETRY ADDRESS
09 02312 040307             STA 0,RLUP
10 02313 020176             LDA 0,C3        ;SET THE PROGRAM MODE
11 02314 040164             STA 0,MODE
12 02315 102000             ADC 0,0
13 02316 040271             STA 0,AECNT
14 02317 040272             STA 0,CWCNT
15 02320 102400             SUB 0,0
16 02321 040103             STA 0,FATAL     ;CLEAR FATAL FLAG
17 02322 020252             LDA 0,SC
18 02323 040306             STA 0,.SC       ;# SECTORS TO XFER
19 02324 006331             GEN
20 02325 000000 WDATA:      0               ;ADDR OF DATA GEN ROUT
21 02326 006335             SETP            ;SET PARAMETERS
22 02327 020174             LDA 0,C400
23 02330 001133             DOAS 0,.DSKP    ;WRITE
24 02331 006300             INTWT           ;WAIT FOR INTERRUPT
25 02332 002274             JMP @RWRET      ;TIMEOUT
26
27 02333 101113             MOVL# 0,0,SNC   ;INTERRUPT RETURN
28 02334 000475             JMP RE.1        ;NO R/W DONE FLAG
29 02335 024227             LDA 1,CSP3      ;077666
30 02336 107404             AND 0,1,SZR
31 02337 000472             JMP RE.1        ;FATAL STATUS
32 02340 024025             LDA 1,C10
33 02341 107404             AND 0,1,SZR
34 02342 000471             JMP RE.2        ;ADDRESS ERROR
35 02343 001433             DIO 0,.DSKP     ;READ MEM ADDR
36 02344 024252             LDA 1,SC/       ;SECTOR COUNT
37 02345 125300             MOVS 1,1        ;WORD COUNT
38 02346 034254             LDA 3,CA        ;STARTING ADDRESS
39 02347 107000             ADD 3,1
40 02350 125400             INC 1,1
41 02351 125400             INC 1,1         ;ADDR+WC SHOULD EQUAL
42 02352 106414             SUB# 3,1,SZR    ;THE ENDING ADDRESS
43 02353 000442             JMP WE.1        ;ERROR
```

```
A 0042  .MAIN
01
02 02354 020252 .W1:    LDA 0,SC          ;FIND ENDING DISK
03 02355 040300          STA 0,CSC         ;ADDRESS
04 02356 006366          JSR @ISCNT        ;AC0=SECT, AC1=HEAD
05 02357 101004          MOV 0,0,SZR       ;IF THE SECT = 0 THE LAST
06 02360 000404          JMP .+4           ;ICR HEAD DID NOT OCCUR
07 02361 020171          LDA 0,LS          ;SECT = LAST SECT
08 02362 124400          NEG 1,1           ;HEAD =1
09 02363 124000          COM 1,1
10 02364 030157          LDA 2,UNIT
11 02365 035026          LDA 3,UNTINS,2    ;FORM THE ENDING DISK
12 02366 125300          MOVS 1,1          ;ADDRESS AS READ BY "DIC"
13 02367 167000          ADD 3,1
14 02370 103120          ADDZL 0,0
15 02371 103120          ADDZL 0,0         ; SECT*4
16 02372 107000          ADD 0,1           ;EXPECTED ADDRESS
17 02373 062433          DIC 0,.DSKP       ;READ ACTUAL
18 02374 106414          SUB# 0,1,SZR
19 02375 000426          JMP WE.2          ;ENDING DISK ADDRESS ERROR
20
21 02376 020252 .W2:     LDA 0,SC          ;ADD WORDS WRITTEN
22 02377 101300          MOVS 0,0          ;TO TOTAL COUNT
23 02400 025075          LDA 1,WDSW+4,2
24 02401 107022          ADDZ 0,1,SZC      ;DOUBLE PRECISION
25 02402 011071          ISZ WDSW,2
26 02403 045075          STA 1,WDSW+4,2
27 02404 020163          LDA 0,FATAL       ;WAS THERE A FATAL ERROR ?
28 02405 101004          MOV 0,0,SZR
29 02406 002274          JMP @RWRET        ;ERROR RETURN
30 02407 020165 GENRET:  LDA 0,LUPSW       ;IS THE LOOP SWITCH ON ?
31 02410 101004          MOV 0,0,SZR
32 02411 002274          JMP @RWRET        ;ERROR RETRUN
33 02412 010274          ISZ RWRET
34 02413 010274          ISZ RWRET
35 02414 002274          JMP @RWRET        ;NORMAL RETURN
36
37 02415 004547 WE.1:    JSR SWCE          ;ENDING MEM ADDR ERROR
38 02416 004512          MSG57             ;IDENTIFIER
39 02417 030157          LDA 2,UNIT
40 02420 011151          ISZ MISC,2        ;COUNT AS MISC TYPE ERROR
41 02421 010163          ISZ FATAL         ;SET FATAL FLAG
42 02422 000732          JMP .W1           ;GO ON
43
44 02423 004541 WE.2:    JSR SWCE          ;ENDING DISK ADDR ERROR
45 02424 004522          MSG67
46 02425 030157          LDA 2,UNIT
47 02426 011151          ISZ MISC,2        ;COUNT AS MISC TYPE ERROR
48 02427 010163          ISZ FATAL         ;SET FATAL FLAG
49 02430 000746          JMP .W2           ;GO ON
```

```
^ 0043  .MAIN
01
02 02431 011151 RE.11:  ISZ MISC,2          ;COUNT AS MISC TYPE ERROR
03 02432 002302          PSTAT              ;PRINT STATUS & ERR RETURN
04
05 02433 000333 RE.21:  HEADER             ;ADDRESS ERROR
06 02434 000324          PCRLF
07 02435 000323          MESSAGE
08 02436 004331          MSG29              ;"ADDRESS ERROR"
09 02437 030157          LDA 2,UNIT
10 02440 010271          ISZ AECNT          ;ADDRESS ERROR COUNTER
11 02441 000410          JMP RE.21          ;2ND FAILURE IN A ROW
12 02442 011121          ISZ ADDER,2        ;FIRST FAILURE
13 02443 020270 RE.20:  LDA 0,EVODN        ;IF TWO COMPUTERS
14 02444 101004          MOV 0,0,SZR        ;DON'T TRY AGAIN
15 02445 002274          JMP @RWRET         ;HEADS MAY HAVE MOVED
16 02446 000324          PCRLF
17 02447 000323          MESSAGE
18 02450 004552          MSG63              ;"TRY AGAIN"
19 02451 020245          LDA 0,RELRAN       ;RESET RAN # GEN FOR
20 02452 040243          STA 0,RANDOM       ;THE RETRY
21 02453 102400          SUB 0,0
22 02454 040162          STA 0,RETRY        ;CLEAR RETRY SWITCH
23 02455 062077          IORST
24 02456 002307          JMP @RLUP          ;DO IT AGAIN
25
26 02457 011125 RE.21:  ISZ PADER,2        ;COUNT PERM ADDR ERRS
27 02460 002274          JMP @RWRET         ;ERROR RETURN
28
29 02461 000477 RE.31:  READS 0            ;CHECK WORD ERROR
30 02462 103120          ADDZL 0,0
31 02463 103122          ADDZL 0,0,SZC      ;IF SW3=1 DON'T PRINT
32 02464 000405          JMP .+5
33 02465 000333          HEADER
34 02466 000324          PCRLF
35 02467 000323          MESSAGE
36 02470 004557          MSG64              ;"CHECK WORD ERROR"
37 02471 030157          LDA 2,UNIT
38 02472 010272          ISZ CWCNT          ;CHECK WORD ERR COUNTER
39 02473 000404          JMP RE.31          ;2ND ERROR IN A ROW
40 02474 011111          ISZ CWER,2         ;FIRST ERROR
41 02475 010162          ISZ RETRY          ;SET RETRY FLAG
42 02476 000403          JMP RE.32
43 02477 011113 RE.31:  ISZ CWERP,2        ;COUNT PERM CHK WD ERRS
44 02500 010163          ISZ FATAL          ;SET FATAL FLAG
45 02501 000004 RE.32:  LDA 0,2A           ;THE CHECK WORD ERROR MAY
46 02502 [illegible]     DIA 1,,DSKP        ;HAVE STOPPED THE DMA
47 02503 [illegible]     SUB 0,1            ;TRANSFER PREMATURELY.
48 02504 000201          LDA 0,C174H        ;FIND THE # OF SECTORS
49 02505 133275          ANDS 1,3,SNR       ;TRANSFERRED AND SAVE IT
50 02506 102400          SUBZL 0,0          ;FOR THE DATA COMPARE ROUTINE.
51 02507 040000          STA 0,.SC          ;IT MUST BE AT LEAST ONE.
52 02510 000000          JMP .R1
```

```
A 0044  .MAIN
01
02 02511 030157 RE.41:   LDA 2,UNIT       ;DATA ERROR
03 02512 020272          LDA 0,CWCNT
04 02513 010273          ISZ DACNT        ;DATA ERROR COUNTER
05 02514 000422          JMP RE.44        ;2ND DATA ERROR
06 02515 010162          ISZ RETRY        ;SET THE RETRY FLAG
07 02516 101404          INC 0,0,SZR      ;FIRST DATA ERROR
08 02517 000403          JMP RE.41        ;DATA AND CHECK WORD ERROR
09 02520 011131          ISZ DATER,2      ;NO CHK WD ERR
10 02521 000530          JMP .R2          ;GO ON
11 02522 024163 RE.41:   LDA 1,FATAL
12 02523 125004          MOV 1,1,SZR
13 02524 000405          JMP RE.43        ;2 CHECK WORD ERRORS
14 02525 015111          DSZ CWER,2       ;ONLY 1 CHECK WORD ERROR
15 02526 000401          JMP .+1          ;SKIPS SOMETIMES
16
17 02527 011135 RE.42:   ISZ CWDE,2       ;COUNT CHK WD & DATA ERRS
18 02530 000521          JMP .R2          ;GO ON
19 02531 015115 RE.43:   DSZ CWERP,2      ;DON'T COUNT FATAL CHK WD ERR
20 02532 000401          JMP .+1
21 02533 102400          SUB 0,0          ;RESET FATAL ERROR FOR NOW
22 02534 040163          STA 0,FATAL
23 02535 000772          JMP RE.42        ;TRY IT A 3RD TIME
24 02536 024163 RE.44:   LDA 1,FATAL      ;FATAL=NON-ZERO IF 2 CHK WD ERR
25 02537 125004          MOV 1,1,SZR
26 02540 000403          JMP RE.45        ;DATA & CHK WD ERR
27 02541 011145          ISZ PDER,2       ;DATA ERR ONLY
28 02542 000404          JMP RE.46        ;2ND ONE IS FATAL
29 02543 015115 RE.45:   DSZ CWERP,2      ;DISCOUNT PERM CHK WD ERR
30 02544 000401          JMP .+1
31 02545 011141          ISZ CWDEP,2      ;COUNT PERM COMBO ERR
32 02546 010163 RE.46:   ISZ FATAL        ;SET THE FATAL FLAG
33 02547 000502          JMP .R2          ;GO ON
```

```
A 0045  .MAIN
01
02 02550 004414 RE.01:   JSR SWCE          ;ENDING MEM ADDR ERR
03 02551 004012          MSG07             ;IDENTIFIER
04 02552 030157          LDA 2,UNIT
05 02553 011151          ISZ MISC,2
06 02554 010163          ISZ FATAL         ;IT'S FATAL
07 02555 000003          JMP .R3
08
09 02556 004400 RE.02:   JSR SWCE          ;ENDING DISK ADDR ERROR
10 02557 004022          MSG07             ;IDENTIFIER
11 02560 030157          LDA 2,UNIT
12 02561 011151          ISZ MISC,2        ;COUNT AS MISC TYPE ERR
13 02562 010163          ISZ FATAL         ;IT'S FATAL
14 02563 000032          JMP .R4
15
16                       ;DA + WC ERROR
17                       ;AC0 = BAD, AC1 = GOOD
18
19 02564 004423 SWCE:    STA 3,SWCRET
20 02565 035400          LDA 3,0,3         ;GET ERROR IDENTIFIER
21 02566 054405          STA 3,SWCE1
22 02567 000310          SAVAC
23 02570 000353          HEADER
24 02571 000324          PCRLF
25 02572 000323          MESSAGE
26 02573 000000 SWCE1:   0
27 02574 000324          PCRLF
28 02575 000323          MESSAGE
29 02576 004404          MSG01             ;"GOOD"
30 02577 024240          LDA 1,SAV1
31 02600 000321          TYPAC1
32 02601 000323          MESSAGE
33 02602 004407          MSG02             ;"BAD"
34 02603 024237          LDA 1,SAV0
35 02604 000321          TYPAC1
36 02605 010402          ISZ SWCRET
37 02606 002401          JMP @SWCRET
38 02607 000000 SWCRET:  0
```

```
A 0046  .MAIN
01
02                      ;READ SUBROUTINE
03                      ;RETURN+3, NORMAL
04                      ;RETURN+1, FATAL ERROR
05
06
07 02010 002025         RGRD            ;RETRY ADDRESS
00 02011 054274 .READ:  STA 3,RWRET
09 02012 020770         LDA 0,.READ-1   ;SET THE RETRY ADDRESS
10 02013 040307         STA 0,RLUP
11 02014 020023         LDA 0,C2        ;SET PROGRAM MODE
12 02015 040164         STA 0,MODE
13 02016 102000         ADC 0,0
14 02017 040271         STA 0,AECNT     ;SET RETRY COUNTERS
15 02020 040272         STA 0,CWCNT
16 02021 040273         STA 0,DACNT
17 02022 102400         SUB 0,0
18 02023 040162         STA 0,RETRY     ;CLEAR RETRY AND
19 02024 040163         STA 0,FATAL     ;FATAL FLAGS.
20 02025 000334 RGRD:   CLRRB           ;CLEAR BUFFER
21 02026 000335         SETP            ;SET PARAMETERS
22 02027 102400         SUB 0,0
23 02030 001133         DOAS 0,.DSKP    ;READ!!
24 02031 000300         INTWT           ;WAIT FOR INTERRUPT
25 02032 002274         JMP @RWRET      ;TIMEOUT
20
27 02033 101113         MOVL# 0,0,SNC   ;INTERRUPT RETURN
28 02034 002501         JMP @IRE1       ;NO R/W DONE FLAG
29 02035 024224         LDA 1,CSP       ;077662
30 02036 107404         AND 0,1,SZR
31 02037 002476         JMP @IRE1       ;FATAL STATUS
32 02040 024020         LDA 1,C10
33 02041 107404         AND 0,1,SZR
34 02042 002474         JMP @IRE2       ;ADDRESS ERROR
35 02043 024024         LDA 1,C4
36 02044 107404         AND 0,1,SZR
37 02045 000014         JMP RE.3        ;CHECK WORD ERROR
38
39 02046 000332 .R1:    CHECK           ;DATA COMPARE
40 02047 000000 RDATA:  0               ;ADDR OF DATA GEN ROUTINE
41 02050 000041         JMP RE.4        ;DATA ERROR
42
43 02051 001433 .R2:    DIB 0,.DSKP     ;READ ENDING ADDRESS
44 02052 024300         LDA 1,.SC       ;# SECTORS XFERRED
45 02053 125000         MOVS 1,1        ;# WORDS
46 02054 020334         LDA 2,SA        ;STARTING MEM ADDR
47 02055 147000         ADD 2,1         ;CORRECT ADDR
48 02056 106412         SUBZ 2,1,SZR
49 02057 000671         JMP RE.5        ;ENDING MEM ADDR ERROR
```

```
▲ 0047  .MAIN
01
02 02600 020306 .R3:    LDA 0,.SC          ;FIND ENDING DISK ADDR
03 02601 040300          STA 0,CSC
04 02602 006550          JSR @IOCNT         ;AC0=SECT AC1=HEAD
05 02603 034306          LDA 3,.SC          ;IF .SC NOT = SC A CHK WD ERR
06 02604 030252          LDA 2,SC           ;TERMINATED THE DATA XFER
07 02605 172414          SUB# 3,2,SZR
00 02606 000400          JMP .+6            ;CHK WD ERR
09 02607 101004          MOV 0,0,SZR        ;IF SECT = 0 NO HEAD ICR
10 02670 000404          JMP .+4            ;OCCURRED AT THE END OF
11 02671 020171          LDA 0,LS           ;THE DATA TRANSFER
12 02672 124400          NEG 1,1            ;SECT = LAST SECTOR ON DISK
13 02673 124000          COM 1,1            ;HEAD = HEAD-1
14 02674 030157          LDA 2,UNIT
15 02675 033020          LDA 3,UNTINS,2
16 02676 125300          MOVS 1,1           ;FORM THE DISK ADDR
17 02677 107000          ADD 3,1            ;WORD AS READ VIA "DIC"
10 02700 103120          ADDZL 0,0
19 02701 103120          ADDZL 0,0          ; SECT*4
20 02702 107000          ADD 0,1
21 02703 020306          LDA 0,.SC          ;# SECT XFERRED
22 02704 030252          LDA 2,SC           ;# SECT SPECIFIED
23 02705 112400          SUB 0,2            ;AC2= # SECT NOT XFERRED
24 02706 020201          LDA 0,C17
25 02707 150400          NEG 2,2
26 02710 113400          AND 0,2            ;FINALLY AC0= EXPECTED
27 02711 147000          ADD 2,1            ;DISK ADDRESS
28 02712 002433          DIC 0,.DSKP        ;READ ACTUAL
29 02713 100414          SUB# 0,1,SZR
30 02714 000642          JMP RE.6           ;ENDING DISK ADDR ERROR
31
32 02715 030157 .R4:     LDA 2,UNIT         ;ADD WORDS READ TO TOTAL
33 02716 020300          LDA 0,.SC
34 02717 101300          MOVS 0,0
35 02720 025103          LDA 1,WDSR+4,2     ;DOUBLE PRECISION
36 02721 107022          ADDZ 0,1,SZC
37 02722 011101          ISZ WDSR,2
30 02723 045105          STA 1,WDSR+4,2
39 02724 020103          LDA 0,FATAL        ;END OF TEST
40 02725 101004          MOV 0,0,SZR        ;WHAT TO DO NOW
41 02726 002274          JMP @RWRET         ;(FATAL) ERROR RETURN
42 02727 020102          LDA 0,RETRY
43 02730 101005          MOV 0,0,SNR
44 02731 002403          JMP 0,+3           ;DO NORMAL RETURN
45 02732 002401          JMP 0,+1           ;TRY AGAIN
40 02733 002443          RE.20
47 02734 002407          GENRET
40
49 02735 002431 IRE1:    RE.1
50 02736 002433 IRE2:    RE.2
```

```
A 0048  .MAIN
01
02                ;       ************************************************
03                ;       ****    GENERAL PURPOSE SUBROUTINES         *****
04                ;       ************************************************
05
06                        ;GET CYL(MAYBE)=HEAD=SECTOR=SECTOR COUNT
07                        ;SET (C)=1 ON RETURN IF CR TERMINATOR
08
09 02737 054433 HSS:    STA 3,HSSRET
10 02740 000326          GETATM          ;GET "SAME", OR HEAD #
11 02741 034255          LDA 3,SAM       ;AC0=# AC1= NAME
12 02742 136415          SUB# 1,3,SNR
13 02743 000424          JMP HSS.1       ;"SAME"
14 02744 125003          MOV 1,1,SNC
15 02745 125004          MOV 1,1,SZR
16 02746 002345          JMP @IGCS       ;OTHER NAMES OR CR ILLEGAL
17 02747 024170          LDA 1,LHD       ;CHECK HEAD LIMIT
18 02750 106437          SUBZ# 0,1,SBN
19 02751 002357          JMP @IQUST      ;LIMIT EXCEDED
20 02752 040253          STA 0,HD        ;HEAD #
21 02753 006326          GETATM          ;GET SECTOR #
22 02754 125003          MOV 1,1,SNC     ;AC0=# AC1=NAME
23 02755 125004          MOV 1,1,SZR
24 02756 002345          JMP @IGCS       ;NAME OR CR TERM ILLEGAL
25 02757 024171          LDA 1,LS        ;CHECK SECT # LIMIT
26 02760 106437          SUBZ# 0,1,SBN
27 02761 002357          JMP @IQUST      ;LIMIT EXCEDED
28 02762 040251          STA 0,SEC       ;SECTOR #
29 02763 006326          GETATM          ;GET # OF SECTORS
30 02764 125004          MOV 1,1,SZR
31 02765 002345          JMP @IGCS       ;NAME ILLEGAL
32 02766 040252          STA 0,SC        ;# OF SECTORS
33 02767 102560 HSS.1:   SUBCL 0,0
34 02770 040256          STA 0,TERM
35 02771 002401          JMP @HSSRET
36 02772 000000 HSSRET:  0
37 02773 000000 HIADD:   0               ;I/O ADDRESS MODIFICATION, HI END
38
39
40                ;       CHECK SWITCH REGISTER ON ERROR.
41                ;       IF SW0 = 0  HALT
42                ;       IF SW2 = 1  DO RETURN +1  (SCOPE LOOP)
43                ;       IF SW2 = 0  DO RETURN +2
44
45 02774 063477 .CSW:    READS 0
46 02775 101120          MOVL 0,0,SNC
47 02776 063677          HALT            ;HALT ON ERROR
48 02777 063677          IORST
49 03000 063477          READS 0         ;READ SWITCHES AGAIN IN CASE
50 03001 101120          MOVZL 0,0       ;THEY WERE CHANGED.
51 03002 103103          ADDL 0,0,SNC
52 03003 173401          INC 3,3,SKP     ;NORMAL RETURN
53 03004 102521          SUBZL 0,0,SKP   ;SET LOOP SWITCH
54 03005 102400          SUB 0,0
55 03006 040155          STA 0,LUPSW
56 03007 001400          JMP 0,3
57
58                        .EOT
```

```
  0049  .MAIN
01
02                          ;SEARCH ROUTINE
03
04                          ;CALL   SEARCH          AC1=NAME
05                          ;       TABLE ADDR.
06                          ;       RETURN - NOT FOUND
07                          ;       RETURN - FOUND, AC2=POINTER
08
09 03010 031400 SRH:        LDA 2,0,3           ;AC2=TABLE ADDRESS
10 03011 021000             LDA 0,0,2
11 03012 101005             MOV 0,0,SNR
12 03013 001401             JMP 1,3             ;NOT FOUND
13 03014 122415             SUB# 1,0,SNR
14 03015 001402             JMP 2,3             ;FOUND
15 03016 151400             INC 2,2
16 03017 000772             JMP SRH+1
17
18
19                          ;SIZE MEMORY, 4K SEGMENTS
20
21 03020 152400 SM:         SUB 2,2
22 03021 020034             LDA 0,C10K
23 03022 113000 SM.1:       ADD 0,2
24 03023 151112             MOVL# 2,2,SZC
25 03024 000405             JMP SM.2            ;32K MACHINE
26 03025 051000             STA 2,0,2           ;STORE
27 03026 025000             LDA 1,0,2           ;READ BACK
28 03027 140403             SUB 2,1,SNR
29 03030 000772             JMP SM.1
30 03031 024212 SM.2:       LDA 1,C177
31 03032 132400             SUB 1,2             ;PROTECT LOADERS
32 03033 050266             STA 2,CMEND
33 03034 020247             LDA 0,LAST
34 03035 112400             SUB 0,2
35 03036 020222             LDA 0,C170K         ;HOW MANY SECTORS WILL MEM
36 03037 143414             AND# 2,0,SZR        ;ALLOW
37 03040 000413             JMP SM.4            ;MAXIMUM, 20 OCTAL
38 03041 020220             LDA 0,C7400
39 03042 113700             ANDS 0,2            ;ROOM FOR 17 OR LESS
40 03043 050267             STA 2,SMAX          ;MAX SECTOR COUNT
41 03044 120400             SUB 1,1
42 03045 103143             MOVOL 1,1           ;GENERATE # SECTORS
43 03046 101024             MOVZR 2,2,SZR       ; MASK
44 03047 000770             JMP .-2
45 03050 103000 SM.3:       MOVS 1,1
46 03051 044270             STA 1,LMSK
47 03052 001400             JMP 0,3
48
49
50 03053 024201 SM.4:       LDA 1,C17
51 03054 121400             INC 1,0
52 03055 040257             STA 0,SMAX          ;MAX SECT CNT=32
53 03056 000772             JMP SM.3
```

```
▲ 0050   .MAIN
01
02                          ;LINE SCAN SUBROUTINE
03                          ;BYTE POINTER IS LINCT
04
05                          ;CALL   GETATM
06                          ;       RETURN = AC0= #
07                          ;                  AC1=NAME
08                          ;                  (C)=1 IF CR DELIMITER
09
10 03057 020510 GA.0:   LDA 0,FIND
11 03060 101005         MOV 0,0,SNR
12 03061 000413         JMP GA.1            ;NO NAME OR # YET
13 03062 020250         LDA 0,LINCT
14 03063 040502         STA 0,SLNCT         ;REMEMBER LINCT
15 03064 040502         STA 0,SRCH          ;SET SEARCH FLAG
16 03065 000407         JMP GA.1
17
18 03066 054474 GATM:   STA 3,GARET
19 03067 102400         SUB 0,0
20 03070 040473         STA 0,ANAM
21 03071 040473         STA 0,ANUM
22 03072 040474         STA 0,SRCH
23 03073 040474         STA 0,FIND
24 03074 030250 GA.1:   LDA 2,LINCT
25 03075 010250         ISZ LINCT
26 03076 151220         MOVZR 2,2
27 03077 021000         LDA 0,0,2
28 03100 024216         LDA 1,C377
29 03101 101002         MOV 0,0,SZC
30 03102 101300         MOVS 0,0
31 03103 123400         AND 1,0             ;(AC0)R = BYTE
32 03104 024205         LDA 1,C40
33 03105 122415         SUB# 1,0,SNR
34 03106 000751         JMP GA.0            ;SPACE DELIMITER
35 03107 024206         LDA 1,C54
36 03110 122415         SUB# 1,0,SNR
37 03111 000740         JMP GA.0            ;COMMA DELIMITER
38 03112 024215         LDA 1,C215
39 03113 122415         SUB# 1,0,SNR
40 03114 000442         JMP EXIT            ;CR DELIMITER
41 03115 024451         LDA 1,SRCH
42 03116 125004         MOV 1,1,SZR
43 03117 000404         JMP EX.1            ;SEARCH FLAG ON
44 03120 030207         LDA 2,C60
45 03121 034213         LDA 3,C67
46 03122 103703         ADCZ# 3,3,SNC
47 03123 113732         ADCZ# 3,2,SZC
48 03124 000412         JMP ASSN            ;NOT #
49 03125 024177         LDA 1,C7
50 03126 123400         AND 1,3
51 03127 024435         LDA 1,ANUM
52 03130 127120         ADDZL 1,1           ;ASSEMBLE OCTAL #
53 03131 125120         MOVZL 1,1
54 03132 123400         ADD 1,3
55 03133 044431         STA 3,ANUM
56 03134 010430         ISZ FIND
57 03135 000737         JMP GA.1            ;GET MORE
```

```
A 0051  .MAIN
01
02 03136 024204 ASSN:   LDA 1,C37       ;ASSEMBLE NAME
03 03137 123400         AND 1,0
04 03140 024423         LDA 1,ANAM
05 03141 000023         LDA 2,C176K     ;3 LETTERS YET ?
06 03142 133494         AND 1,2,SZR
07 03143 000731         JMP GA.1        ;YES, IGNORE THE REST
08 03144 127120         ADDZL 1,1
09 03145 127120         ADDZL 1,1       ;5 LEFT
10 03146 125120         MOVZL 1,1
11 03147 123000         ADD 1,0
12 03150 040413         STA 0,ANAM
13 03151 010410         ISZ FIND
14 03152 000722         JMP GA.1        ;GET MORE
15
16 03153 020412 EX.1:   LDA 0,SLNCT
17 03154 040250         STA 0,LINCT
18 03155 101021         MOVZ 0,0,SKP    ;CLEAR CARRY
19 03156 101040 EXIT:   MOVO 0,0        ;SET CARRY
20 03157 020405         LDA 0,ANUM
21 03160 024403         LDA 1,ANAM
22 03161 002401         JMP @GARET
23
24 03162 000000 GARET:  0
25 03163 000000 ANAM:   0
26 03164 000000 ANUM:   0
27 03165 000000 SLNCT:  0
28 03166 000000 SRCH:   0
29 03167 000000 FIND:   0
```

```
A 0052  .MAIN
01
02                          ;PRINT HEADER
03
04 03170 010155 HED:    ISZ HSW          ;PRINT HEADER ONLY ONCE
05 03171 001400          JMP 0,3          ;FOR EACH COMPLETE TEST
06 03172 054440          STA 3,HRET
07 03173 020440          LDA 0,HRET+1     ;THE MODE DETERMINES
08 03174 030164          LDA 2,MODE       ;THE TYPE OF OPERATION
09 03175 113000          ADD 0,2          ;IN PROGRESS
10 03176 021000          LDA 0,0,2
11 03177 040404          STA 0,HED.1
12 03200 006324          PCRLF
13 03201 006324          PCRLF
14 03202 006323          MESSAGE
15 03203 000000 HED.1:   0                ;"RECAL OR SEEK, ETC.
16 03204 006323          MESSAGE
17 03205 004447          MSG44            ;CYL=
18 03206 030157          LDA 2,UNIT
19 03207 025051          LDA 1,CYLT,2
20 03210 006322          TYPZ1
21 03211 006323          MESSAGE
22 03212 004452          MSG45            ;HEAD=
23 03213 024253          LDA 1,HD
24 03214 006322          TYPZ1
25 03215 006323          MESSAGE
26 03216 004455          MSG46            ;SECT=
27 03217 024251          LDA 1,SEC
28 03220 006322          TYPZ1
29 03221 006323          MESSAGE
30 03222 004460          MSG47            ;#SECT=
31 03223 024252          LDA 1,SC
32 03224 006322          TYPZ1
33 03225 006323          MESSAGE
34 03226 004240          MSG17            ;UNIT:
35 03227 024157          LDA 1,UNIT
36 03230 006322          TYPZ1
37 03231 002401          JMP @HRET
38 03232 000000 HRET:    0
39 03233 003234          .+1
40 03234 004163          MSG2             ;0  RECAL
41 03235 004167          MSG3             ;1  SEEK
42 03236 004413          MSG32            ;2  READ
43 03237 004416          MSG33            ;3  WRITE
```

```
A 0053   .MAIN
01
02
03
04 03240 102041 C:      SUBOL 0,0,SKP     ;CHECK DATA...
05 03241 102020 G:      SUBZL 0,0         ;GENERATE DATA...
06 03242 040614         STA 0,@STGC       ;C(CA)=START ADDRESS
07 03243 034611         STA 3,GRET        ;C(SC)  =NUMBER OF SECTORS
08 03244 021400         LDA 0,0,3         ;WORD FOLLOWING THE CALL
09 03245 040612         STA 0,PATT        ;DEFINES THE PATTERN.
10 03246 010600         ISZ GRET
11 03247 020201         LDA 0,VARST
12 03250 040204         STA 0,VARPT
13 03251 020254         LDA 0,CA
14 03252 040020         STA 0,IDX0
15 03253 014020         DSZ IDX0
16 03254 020300         LDA 0,.SC
17 03255 101300         MOVS 0,0
18 03256 100400         NEG 0,0
19 03257 040601         STA 0,GWC
20 03260 014470         DSZ @STGC         ;GEN/CHECK FIRST SWITCH
21 03261 000440         JMP CC
22 03262 006470 GG:     JSR @PATT         ;GENERATE DATA
23 03263 042020         STA 0,@IDX0       ;STORE IN MEMORY
24 03264 010474         ISZ GWC
25 03265 000775         JMP GG
26 03266 002400         JMP @GRET
27
28
29 03267 070477 CKER:   READS 2           ;IF SW3=1 DON'T PRINT ERRORS
30 03270 153120         ADDZL 2,2
31 03271 153122         ADDZL 2,2,SZC
32 03272 000432         JMP CK4
33 03273 054402         STA 3,CKRET       ;A ERROR DETECTED
34 03274 044405         STA 1,BAD
35 03275 040405         STA 0,GOOD
36 03276 010400         ISZ @STGC
37 03277 000405         JMP CK1           ;DON'T PRINT HEADER
38 03300 006303         HEADER            ;"ADDR GOOD BAD WORD
39 03301 006324         PCRLF             ;ETC.
40 03302 006323         MESSAGE
41 03303 004301         MSG22
42 03304 030402 CK1:    LDA 2,@STGC
43 03305 170120         ADCZL 3,3
44 03306 157006         ADDZ# 2,3,SZC
45 03307 002346         JMP @CKRET        ;EXIT IF>2 ERRORS
```

```
A 0054  .MAIN
01
02 03310 000324          PCRLF
03 03311 024020          LDA 1,IDX0
04 03312 006322          TYPZ1              ;PRINT ADDRESS
05 03313 024447          LDA 1,GOOD
06 03314 006321          TYPAC1             ;GOOD DATA
07 03315 024444          LDA 1,BAD
08 03316 006321          TYPAC1             ;BAD DATA
09
10 03317 020254          LDA 0,CA
11 03320 024020          LDA 1,IDX0
12 03321 106400          SUB 0,1
13 03322 006322          TYPZ1              ;WORD#
14 03323 002432          JMP @CKRET
15
16 03324 010432 CK4:     ISZ FSTGC          ;COUNT ERRORS
17 03325 000401          JMP .+1
18 03326 001400          JMP 0,3            ;RETURN
19
20 03327 006430 CC:      JSR @PATT          ;CHECK THE DATA
21 03330 020020          LDA 1,@IDX0
22 03331 106414          SUB# 0,1,SZR
23 03332 004735          JSR CKER           ;CHECK ERROR
24 03333 010425          ISZ GWC
25 03334 000773          JMP CC
26 03335 010421          ISZ FSTGC          ;ANY ERRORS ??
27 03336 000402          JMP .+2
28 03337 000413          JMP CC1            ;NO, EXIT
29 03340 060477          READS 0            ;IF SW3=1 DON'T PRINT ERRORS
30 03341 103120          ADDZL 0,0
31 03342 103122          ADDZL 0,0,SZC
32 03343 002411          JMP @GRET
33 03344 006324          PCRLF              ;YES, PRINT THE
34 03345 024411          LDA 1,FSTGC        ;TOTAL # OF ERRORS
35 03346 006322          TYPZ1
36 03347 006323          MESSAGE
37 03350 004506          MSG56              ;"ERRORS"
38 03351 002403          JMP @GRET
39
40 03352 010402 CC1:     ISZ GRET           ;NORMAL RETURN + 2
41 03353 002401          JMP @GRET
42
43 03354 000000 GRET:    0
44 03355 000000 CKRET:   0
45 03356 000000 FSTGC:   0
46 03357 000000 PATT:    0
47 03360 000000 CA:      0
48 03361 000000 BAD:     0
49 03362 000000 GOOD:    0
```

```
A 0035  .MAIN
01
02                              ;OPERATOR SPECIFIED WORDS
03
04 03303 054413 VAR.0:  STA 3,VARET
05 03304 010204         ISZ VARPT
06 03305 030204         LDA 2,VARPT
07 03306 034203         LDA 3,VARSP
08 03307 172015         ADC# 3,2,SNR
09 03370 000403         JMP VAR.1       ;END INPUT
10 03371 021000         LDA 0,0,2
11 03372 002404         JMP @VARET
12 03373 030201 VAR.1:  LDA 2,VARST
13 03374 050204         STA 2,VARPT
14 03375 000767         JMP VAR.0+1
15
16 03376 000000 VARET:  0
17
18
19
20                              ;RANDOM NUMBER GENERATOR
21
22 03377 054431 RAN:    STA 3,.UD03     ;GENERATE A RANDOM
23 03400 050427         STA 2,.UD02
24 03401 044425         STA 1,.UD01
25 03402 020243         LDA 0,RANDOM    ;NUMBER IN AC0
26 03403 004410         JSR .UD60
27 03404 034420         LDA 3,.UD20
28 03405 103000         ADD 3,0
29 03406 040243         STA 0,RANDOM    ;STORE NEW VALUE.
30 03407 111100         MOVL 0,2
31 03410 030417         LDA 2,.UD02
32 03411 024415         LDA 1,.UD01
33 03412 002410         JMP @.UD03
34
35 03413 024420 .UD60:  LDA 1,.UD21     ;RANDOM CONTINUED
36 03414 044410         STA 1,.UD10
37 03415 105120         MOVZL 0,1
38 03416 125120         MOVZL 1,1
39 03417 014412         DSZ .UD10
40 03420 000776         JMP .-2
41 03421 107000         ADD 0,1
42 03422 105120         MOVZL 1,1
43 03423 125120         MOVZL 1,1
44 03424 103000         ADD 1,0
45 03425 111100         JMP 0,0
46 03426 000000 .UD11:  0
47 03427 000000 .UD02:  0
48 03430 000000 .UD03:  0
49 03431 000000 .UD10:  0
50 03432 000001 .UD20:  00001
51 03433 000010 .UD21:  10
```

```
A 0056  .MAIN
01                ; BINARY TO DECIMAL ASCII CONVERT
02                ; CONVERTS A DOUBLE PRECISION, TWO'S COMPLEMENT NUMBER
03                ; TO AN ASCII DECIMAL CHARACTER STRING
04
05                ; INPUT:          D IN AC1, AC2 (HIGH, LOW)
06
07                ; OUTPUT:         ASCII CHARACTER STRING, TERMINATED BY A
08                ;                 NULL WORD.
09                ;                 CHARACTERS PASSED RIGHT ADJUSTED,
10                ;                 BIT 0 = 0, IN AC0 TO USER
11                ;                 ROUTINE WHOSE ADDRESS MUST BE
12                ;                 STORED IN LOCATION 41 OF PAGE 0
13
14                ;                 STRING OF FORM:
15                ;                         +NNNNNNNNNN(NULL)
16                ;                 OR      -NNNNNNNNNN(NULL)
17
18                ; CALLING SEQUENCE
19                ;       JSR       .DBD
20                ;       RETURN
21
22                ; DESTROYED:       AC1, AC2, AC3, CARRY
23                ; UNCHANGED:       AC0
24
25 03434 054456 .DBD:   STA 3,.FD03       ; SAVE RETURN
26 03435 040454         STA 0,.FD00       ; SAVE AC0
27 03436 020002         LDA 0,.FD30       ; POINT TO HIGH ORDER POWER IN
28                                        ; TABLE
29 03437 040506         STA 0,.FD12
30 03440 176400         SUB 3,3
31 03441 054452         STA 3,DIGIT
32 03442 044500 .FD99:  STA 1,.FD10       ; SAVE ABS(NUMBER)
33 03443 050500         STA 2,.FD10+1
34 03444 034447         LDA 3,DIGIT
35 03445 175004         MOV 3,3,SZR
36 03446 000041         JSR 0,.FD40       ; PUT OUT SIGN OR DIGIT
37 03447 024473         LDA 1,.FD10       ; RESTORE ABS(NUMBER)
38 03450 030473         LDA 2,.FD10+1
39 03451 020477         LDA 0,.FD22       ; GET OCTAL 57
40 03452 040472         STA 0,.FD11       ; COUNT IT UP IN STORAGE
41 03453 034472         LDA 3,.FD12       ; CURRENT POINTER TO POWER OF
42                                        ; 10 TABLE
43
44
45 03454 021401 .FD98:  LDA 0,1,3         ; LOW ORDER WORD
46 03455 101000         MOVZR# 0,0,SNR
47 03456 000452         ISZ DIGIT
48 03457 101000         MOV 0,0,SNR       ; TEST FOR END OF TABLE
49 03460 000430         JMP .FD97         ; DONE
50 03461 112402         SUBZ 0,2
51 03462 021400         LDA 0,0,3         ; HIGH ORDER WORD
52 03463 101000         MOV 0,0,SNC
53 03464 101001         ADC 0,1,SNR
54 03465 102400         SUB 0,1
55 03466 040443         ISZ .FD11         ; COUNT UP DIGIT
56 03467 103110         MOVL# 1,1,SNC     ; TEST FOR <0
57 03470 000764         JMP .FD98         ; KEEP SUBTRACTING
```

```
A 0057  .MAIN
01
02 03471 021401          LDA 0,1,3          ; RESTORE POSITIVE VALUE
03 03472 113022          ADDZ 0,2,SZC
04 03473 125400          INC 1,1
05 03474 021400          LDA 0,0,3
06 03475 107000          ADD 0,1
07 03476 175400          INC 3,3            ; BUMP AC3 TO NEXT TABLE ENTRY
00 03477 175400          INC 3,3
09 03500 054445          STA 3,.FD12
10 03501 020443          LDA 0,.FD11        ; GET DIGIT
11 03502 034207          LDA 3,C60
12 03503 116414          SUB# 0,3,SZR
13 03504 010407          ISZ DIGIT
14 03505 000733          JMP .FD99          ; PUT IT OUT
15
16 03506 000323 .FD97:   JSR @IMESS
17 03507 004313          M8G23              ;"TAB"
18
19 03510 002402          JMP @.FD03         ; RETURN
20
21 03511 000000 .FD00:   0                  ; SAVE AC0
22 03512 000000 .FD03:   0                  ; SAVE RETURN
23 03513 000000 DIGIT:   0
24
25 03514 035032 .FD05:   35032              ; 10**9
26 03515 145000          145000
27 03516 002760          2760               ; 10**8
28 03517 100400          100400
29 03520 000230          230                ; 10**7
30 03521 113200          113200
31 03522 000017          17                 ; 10**6
32 03523 041100          41100
33 03524 000001          1                  ; 10**5
34 03525 103240          103240
35       000012          .RDX 10
36 03526 000000          0                  ; 10**4
37 03527 023420          10000
38 03530 000003          3                  ; 10**3
39 03531 001750          1750
40 03532 000000          0                  ; 10**2
41 03533 000144          144
42 03534 000000          0                  ; 10**1
43 03535 000012          12
44 03536 000000          0                  ; 10**0
45 03537 000001          1
46 03540 03[illegible] .FD03:   .FD05       ; POINTER TO CONVERSION TABLE
47 03541 [illegible]     0                  ; END OF TABLE INDICATOR
48       000010          .RDX 8
49
50       000002 .FD10:   .BLK 2              ; SAVE CURRENT DOUBLE WORD
51 03544 [illegible] .FD11:   0             ; COUNT OF DIGIT WORD
52 03545 000015 .FD12:   15                 ; POINTER TO POWER OF TEN ENTRY
53
54 03546 000133 .FD99:   "[illegible]       ; ASCII "["
55 03547 [illegible] .FD01:   "[illegible]  ; ASCII "["
56 03550 [illegible] .FD02:   57            ; ASCII "0"+9
57       000041 .FD[illegible]=41           ; PAGE 3 PUT CHARACTER ADDRESS
```

```
0058   .MAIN
01                 ;TTO NON INTERRUPT PACKAGE
02                 ;"MESS" PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLR
03                 ;"CHAR" PRINTS ASCII CHARACTOR, C(0)R,C(0)L MUST BE 0
04                 ;WILL RETURN +2 IF C(0)R=0,CORRECTS THE PARITY,33 SIMULATE
05                 ;"TYPE" PRINTS C(0)R. MUST HAVE PROPER PARITY. RETURN IS
06                 ;TO CALL+1.REPLACE THIS ROUTINE WITH INTERRUPT TYPE IF DESIRED.
07                 ;"CRLF" PRINTS A CARRIAGE RETURN
08                 ;"POCT" PRINTS C(1) IN OCTAL FOLLOWED BY A TAB
09                 ;"PDEC" PRINTS C(1) IN DECIMAL,LEADING ZEROS SUPPRESSED,
10                 ;FOLLOWED BY A TAB.
11 03551 054566 MESS:   STA 3,MESSR       ;PRINT A TEXT MESSAGE
12 03552 070477         READC 2
13 03553 153102         ADDL 2,2,SZC      ;NO PRINT IF SW1=1
14 03554 001401         JMP 1,3
15 03555 010562         ISZ MESSR
16 03556 031400         LDA 2,0,3         ;C(2) POINTS TO MESSAGE
17 03557 024216         LDA 1,C377        ;A 8 BIT MASK
18 03560 021000 MES.1:  LDA 0,0,2         ;C(2)=DATA WORD
19 03561 125112         MOVL# 1,1,SZC
20 03562 123701         ANDS 1,0,SKP
21 03563 123401         AND 1,0,SKP       ;C(0)=DATA CHARACTOR RIGHT
22 03564 151400         INC 2,2           ;INC TO NEXT WORD
23 03565 124000         COM 1,1           ;FLIP MASK
24 03566 004470         JSR CHAR.         ;PRINT
25 03567 000771         JMP MES.1         ;ANOTHER
26 03570 063511         SKPBZ TTO
27 03571 000777         JMP .-1
28 03572 060211         NIOC TTO
29 03573 002544         JMP @MESSR        ;LAST
30
31 03574 102401 ZOCT:   SUB 0,0,SKP
32 03575 020207 POCT:   LDA 0,C00
33 03576 030437         LDA 2,OCTAB       ;PRINT C(1) IN OCTAL
34 03577 000403         JMP .+3
35 03600 030445 PDEC:   LDA 2,DECTB       ;PRINT C(1) IN DECIMAL
 0 03601 102400         SUB 0,0
37 03602 054453         STA 3,RADRET      ;BOTH ENTRYS PRINT NUMBER
38 03603 074477         READS 3
39 03604 177102         ADDL 3,3,SZC      ;NO PRINT IF SW1=1
40 03605 002450         JMP @RADRET
41 03606 040440         STA 0,ZSUPP       ;THEN TAB TO NEXT POSITION
 2 03607 050401         STA 2,.+1
43 03610 000000 DECOCT: 0                 ;A"LDA 2,TABLE" INSTRUCTION
44 03611 010777         ISZ .-1
45 03612 034433         LDA 3,RADRET      ;SETUP "TAB" AT END
46 03613 030413         LDA 2,CMTAB
47 03614 151000         MOV 2,2,SNR       ;IF TABLE ENTRY=0
 3 03615 004431         JMP CHAR.         ;EXIT WITH TAB
49 03616 034430         LDA 3,ZSUPP       ;ZEROS SUPPRESS STUF
50 03617 102400         SUB 0,0
51 03620 142012 DECOT:  SUBL# 2,1,SZC
52 03621 014433         JMP DECP
53 03622 152400         SUB 3,1           ;FORM THE DIGIT
 4 03623 034427         LDA 3,C60
55 03624 101100         INC 0,0
56 03625 100775         JMP DECOT
```

```
A 0000   .MAIN

   03026 151230 DECP:    MOVZR# 2,2,SNR
   03027 034207          LDA 3,C60
   03030 054424          STA 3,ZSUPP       ;C(0)=DIGIT
   03031 163000          ADD 3,0           ;MAKE ASCII
   03032 175004          MOV 3,3,SZR
   03033 004423          JSR CHAR.         ;PRINT
   03034 000754          JMP DECOCT        ;GET NEXT DIGIT


   03035 030426 OCTAB:   LDA 2,.+1+.-DECOCT
   03036 100000          100000
   03037 010000          10000
   03040 001000          1000
   03041 000100          100
   03042 000010          10
   03043 000001          1
   03044 000000          0

   03045 030436 DECTB:   LDA 2,.+1+.-DECOCT
         000012 .RDX 10
   03046 023420          10000
   03047 001750          1000
   03050 000144          100
   03051 000012          10
   03052 000001          1
   03053 000000          0
         000010 .RDX 8

   03054 000000 ZSUPP:   0
   03055 000000 RADRET:  0
```

```
▲ 0060   .MAIN
01
02 03656 054404 CHAR.:  STA 3,CHRET      ;PRINT C(0) RIGHT
03 03657 101325         MOVZS 0,0,SNR    ;RETURN +2 IF NULL
04 03660 001401         JMP 1,3
05 03661 040452         STA 0,CHSAV
06 03662 176000         ADC 3,3          ;COMPUTE THE PARITY
07 03663 117000         ADD 0,3
08 03664 163404         AND 3,0,SZR
09 03665 000775         JMP .-3
10 03666 176060         SUBCR 3,3        ;COMBIND PARITY WITH CHAR
11 03667 020444         LDA 0,CHSAV
12 03670 163300         ADDS 3,0
13
14 03671 034440 CHAR1:  LDA 3,CHTAB      ;IS THIS A TAB
15 03672 116415         SUB# 0,3,SNR
16 03673 000407         JMP .+7          ;YES
17 03674 004444         JSR TYPE         ;NO PRINT IT
18 03675 000413         JMP CHAR2+1      ;EXIT
19
20 03676 020436         LDA 0,CHORZ      ;SIMULATE A TAB
21 03677 034200         LDA 3,C12        ;VIA 1 TO 10 SPACES
22 03700 162420         SUBZ 3,0,SEZ
23 03701 000777         JMP .-1
24 03702 101005         MOV 0,0,SNR
25 03703 000404         JMP CHAR2
26 03704 020431         LDA 0,CH240
27 03705 004433         JSR TYPE
28 03706 000770         JMP .-10
29 03707 040425 CHAR2:  STA 0,CHORZ
30 03710 063511         SKPBZ TTO
31 03711 000777         JMP .-1
32 03712 060211         NIOC TTO
33 03713 002417         JMP @CHRET
```

```
▲ 0001  .MAIN
01
02 03714 034422 CRLF:   STA 3,CRLFR
03 03715 060477         READS 0
04 03716 103100         ADDL 0,0,SZC     ;NO PRINT IF SW1=1
05 03717 000407         JMP CRLF1
06 03720 020215         LDA 0,C215
07 03721 004735         JSR CHAR.        ;PRINT CARRIAGE AND LF
08 03722 020214         LDA 0,C212
09 03723 004733         JSR CHAR.
10 03724 020216         LDA 0,C377       ;PRINT RUB
11 03725 004731         JSR CHAR.
12 03726 102400 CRLF1:  SUB 0,0
13 03727 040405         STA 0,CHORZ      ;CLEAR HORZ POSISTION
14 03730 002406         JMP @CRLFR       ;EXIT
15
16 03731 000011 CHTAD:  11
17 03732 000000 CHRET:  0
18 03733 000000 CHSAV:  0
19 03734 000000 CHORZ:  0
20 03735 000240 CH240:  240
21 03736 000000 CRLFR:  0
22
23 03737 000000 HESSR:  0
24 03740 054411 TYPE:   STA 3,TYPRET     ;TYPE THE C(0)R IF
25 03741 010773         ISZ CHORZ
26 03742 074477         READS 3
27 03743 177112         ADDLE# 3,3,SZC   ;AC1=1 DON'T TYPE
28 03744 002405         JMP @TYPRET
29 03745 063511         SKPBZ TTO
30 03746 000777         JMP .-1
31 03747 061111         DOAS 0,TTO
32 03750 002401         JMP @TYPRET
33 03751 000000 TYPRET: 0
34
35
36
37                      ;SAVE AC0,1,2
38 03752 040037 SAC:    STA 0,SAV0
39 03753 044240         STA 1,SAV1
40 03754 050241         STA 2,SAV2
41 03755 001400         JMP 0,3
42
43                      ;RESTORE AC0,1,2
44 03756 020037 STAC:   LDA 0,SAV0
45 03757 024240         LDA 1,SAV1
46 03760 030241         LDA 2,SAV2
47 03761 001400         JMP 0,3
48
```

```
A 0062  .MAIN
01                      ;TELETYPE INPUT ROUTINE, NON INTERRUPT
02                      ;
03                      ;CALL   INPUT
04                      ;       ADDR OF BYTE POINTER
05                      ;       RETURN = CR ONLY
06                      ;       RETURN = NORMAL
07                      ;INPUT IS STORED R-L IN 7 BIT ASCII
08                      ;INPUT IS TERMINATED BY CR, (215) IS
09                      ;STORED.  LINE FEED ECHOS CR-LF
10                      ;NO DATA STORED, INPUT CONTINUES.
11
12 03702 054514 INP:    STA 3,INPRET
13 03703 023400         LDA 0,00,3      ;GET BYTE POINTER
14 03704 040511         STA 0,BASE
15 03705 040512         STA 0,BPTR
16 03706 060210         NIOC TTI
17 03707 020512         LDA 0,M115.
18 03710 040510         STA 0,CHCNT
19 03711 063610 TTWAIT: SKPDN TTI       ;WAIT FOR INPUT
20 03712 000777         JMP .-1
21 03713 064010         DIAC 1,TTI      ;READ INPUT CHAR
22 03714 020210         LDA 0,C377
23 03715 106415         SUB# 0,1,SNR
24 03716 000442         JMP RUB         ;RUB OUT
25 03717 020200         LDA 0,C12
26 04000 106414         SUB# 0,1,SZR
27 04001 000403         JMP .+3
28 04002 000324         PCRLF           ;LINE FEED TYPED
29 04003 000760         JMP TTWAIT
30 04004 060111         DOAS 1,TTO      ;ECHO CHAR
31 04005 030472         LDA 2,BPTR
32 04006 010471         ISZ BPTR
33 04007 034215         LDA 3,C215      ;CR CODE
34 04010 106405         SUB 1,3,SNR
35 04011 000456         JMP CRCOD       ;CR TYPED
36 04012 020212         LDA 0,C177      ;7 BIT MASK
37 04013 151223 INP.01: MOVZR 2,2,SNC
38 04014 107401         AND 0,1,SKP     ;STORE INTO RH
39 04015 107701         ANDS 0,1,SKP
40 04016 000403         JMP .+3
41 04017 021000         LDA 0,0,2
42 04020 107000         ADD 0,1
43 04021 041000         STA 1,0,2       ;STORE BYTE
44 04022 010400         ISZ CHCNT
45 04023 000422         JMP .+2
46 04024 000435         JMP CMPL
47 04025 151124         MOV 3,3,SZR     ;ACSHU IF CR TYPED
48 04026 000740         JMP TTWAIT
49 04027 010407         ISZ INPRET      ;ALL DONE
50 04030 000324 INP.1:  PCRLF
51 04031 020400         LDA 0,BASE
52 04032 040404         STA 0,LINCT
53 04033 010400         ISZ INPRET
54 04034 063611         SKPDZ TTO       ;WAIT FOR ECHO TO
55 04035 000777         JMP .-1         ;FINISH
56 04036 063611         NIOC TTO        ;CLEAR INTERRUPT
57 04037 002437         JMP @INPRET
```

```
A 0003   .MAIN
01
02 04040 020441 RUB:    LDA 0,M113.
03 04041 024037          LDA 1,CHCNT
04 04042 122403          SUB 1,0,SNR
05 04043 000720          JMP TTWAIT      ;NOTHING TO RUB
06 04044 014433          DSZ BPTR
07 04045 014433          DSZ CHCNT
08 04046 030431          LDA 2,BPTR
09 04047 151220          MOVZR 2,2
10 04050 021000          LDA 0,0,2       ;GET BYTE
11 04051 101003          MOV 0,0,SNC
12 04052 000405          JMP RUB1
13 04053 024210          LDA 1,C377
14 04054 107400          AND 0,1         ;SAVE RH ONLY
15 04055 045000          STA 1,0,2
16 04056 101300          MOVS 0,0        ;ECHO LH (RUBBED OUT)
17 04057 001111 RUB1:    DOAS 0,TTO      ;ECHO RUBBED CHAR
18 04060 000711          JMP TTWAIT
19
20 04061 000324 OVFL:    PCRLF           ;LINE OVERFLOW
21 04062 000323          MESSAGE
22 04063 004271          MSG21           ;"INPUT OVERFLOW"
23 04064 014413          DSZ BPTR        ;BACKUP BYTE POINTER
24 04065 014413          DSZ CHCNT
25 04066 000703          JMP TTWAIT
26
27 04067 020400 CRCOD:   LDA 0,BASE
28 04070 112413          SUB# 0,2,SNR
29 04071 000737          JMP INP.1       ;YES
30 04072 020210          LDA 0,C377
31 04073 170400          SUB 3,3
32 04074 000717          JMP INP.0
33
34 04075 000000 BASE:    0
35 04076 000000 INPRET:  0
36 04077 000000 BPTR:    0
37 04100 000000 CHCNT:   0
38 04101 177715 M115.:   -115.
```

```
A 0064  .MAIN
01                          ;DISK COMMAND TABLE
02
03 04102 044241 DCT:        044241          ;READ
04 04103 057111             057111          ;WRITE
05 04104 040245             040245          ;SEEK
06 04105 044243             044243          ;RECALIBRATE
07 04106 030757             030757          ;LOOP
08 04107 010254             010254          ;DELAY
09 04110 000000             0
10
11                          ;DISPATCHES
12
13 04111 002127 DCT.1:      RE
14 04112 002143             WT
15 04113 002160             SK
16 04114 002175             RCL
17 04115 002203             LUP
18 04116 002210             DLAY
19
20                          ;DATA TABLE
21
22 04117 014032 DNT:        14032           ;FLOATING ZERO
23 04120 014017             14017           ;FLOATING ONE
24 04121 044056             044056          ;RANDOM
25 04122 002017             2017            ;ALL 1'S
26 04123 002032             2032            ;ALL ZEROS
27 04124 040064             40064           ;110110 PAT
28 04125 000000             0
29
30                          ;DISPATCHES
31
32 04126 004147 DNT.1:      PLZ
33 04127 004142             PL1
34 04130 003377             RAN
35 04131 004133             ONES
36 04132 004134             ZEROS
37 04133 004140             PAT1
```

```
A 0006  .MAIN
01
02 04134 102401 ZERO0:  SUB 0,0,SKP
03 04135 102000 ONE0:   ADC 0,0
04 04136 001400         JMP 0,3
05
06 04137 155555         155555
07 04140 020777 PAT1:   LDA 0,.-1
00 04141 001400         JMP 0,3
09
10 04142 020030 FL1:    LDA 0,FLO1
11 04143 101225         MOVZR 0,0,SNR
12 04144 101240         MOVOR 0,0
13 04145 040030         STA 0,FLO1
14 04146 001400         JMP 0,3
15
16 04147 020037 FLZ:    LDA 0,FLOZ
17 04150 101243         MOVOR 0,0,SNC
18 04151 102220         ADCZR 0,0
19 04152 040037         STA 0,FLOZ
20 04153 001400         JMP 0,3
```

```
                                  ;ENDING STATUS
                        MSG1:     .TXTE |ENDING STATUS|
    04154 047305
    04155 144504
    04156 043516
    04157 051640
    04160 040724
    04161 052724
    04162 000123
                                  ;RECAL
                        MSG2:     .TXTE |RECAL     |
    04163 142722
    04164 040703
    04165 004714
    04166 000000
                                  ;SEEK
                        MSG3:     .TXTE |SEEK      |
    04167 142523
    04170 045705
    04171 000011
                                  ;INTERRUPT FROM DEVICE
                        MSG9:     .TXTE |INTERRUPT FROM DEVICE |
    04172 047311
    04173 142724
    04174 151322
    04175 050125
    04176 120324
    04177 151306
    04200 040717
    04201 042240
    04202 053305
    04203 141711
    04204 120305
    04205 000000
                                  ;DISK
                        MSG13:    .TXTE |DISK                        |
    04206 144504
    04207 045523
    04210 004411
    04211 000011
                                  ;SEEK ERRORS
                        MSG14:    .TXTE |SEEK ERRORS                 |
    04212 142523
    04213 045705
    04214 [illegible]
    04215 [illegible]
    04216 [illegible]
    04217 004523
    04220 000011
                                  ;TOTAL SEEKS
                        MSG15:    .TXTE |TOTAL SEEKS                 |
    04221 [illegible]
    04222 [illegible]
    04223 [illegible]
    04224 [illegible]
    04225 [illegible]
    04226 004523
    04227 000011
```

0067 .MAIN

```
01                            ;NO READY UNITS
02                   MSG16:   .TXTE |NO READY UNITS|
    04230 147016
    04231 151240
    04232 040705
    04233 054504
    04234 052040
    04235 144516
    04236 051724
    04237 000000
03                            ;UNIT:
04                   MSG17:   .TXTE |UNIT: |
    04240 047125
    04241 152311
    04242 120072
    04243 000000
05                            ;DATA:
06                   MSG18:   .TXTE |DATA: |
    04244 040504
    04245 040724
    04246 120072
    04247 000000
07                            ;COMMAND STRING:
08                   MSG19:   .TXTE |COMMAND STRING: |
    04250 147703
    04251 046515
    04252 047101
    04253 120104
    04254 152123
    04255 144722
    04256 043516
    04257 120072
    04260 000000
09                            ;16 NUMBERS MAX.
10                   MSG20:   .TXTE |16 NUMBERS MAX.|
    04261 033261
    04262 047240
    04263 040525
    04264 142502
    04265 051722
    04266 046040
    04267 154101
    04270 000056
11                            ;INPUT OVERFLOW
12                   MSG21:   .TXTE |INPUT OVERFLOW|
    04271 047011
    04272 [illegible]
    04273 [illegible]
    04274 000017
    04275 [illegible]
    04276 [illegible]
    04277 100717
    04300 [illegible]
13                            ;ADDR    GOOD     BAD      WORD
14                   MSG22:   .TXTE |ADDR      GOOD     BAD       WORD|
    04301 [illegible]
    04302 [illegible]
    04303 043411
    04304 147717
```

```
0060   .MAIN
 04305 004504
 04306 040502
 04307 004504
 04310 147727
 04311 042322
 04312 000000
01                      ; TAB
02              MSG23:  .TXTE | |
 04313 000011
03                      ;WORDS WRITTEN
04              MSG25:  .TXTE |WORDS WRITTEN          |
 04314 147727
 04315 042322
 04316 120123
 04317 151327
 04320 152311
 04321 142724
 04322 004516
 04323 000011
05                      ;WORDS READ
06              MSG26:  .TXTE |WORDS READ             |
 04324 147727
 04325 042322
 04326 120123
 04327 142722
 04330 042101
 04331 004411
 04332 000000
07                      ;CHECK WORD ERRORS
08              MSG27:  .TXTE |CHECK WORD ERRORS                |
 04333 044303
 04334 141705
 04335 120113
 04336 147727
 04337 042322
 04340 142040
 04341 151322
 04342 151317
 04343 004523
 04344 000011
09                      ;PERM CHECK WORD ERRORS
10              MSG28:  .TXTE |PERM CHECK WORD ERRORS   |
 04345 142520
 04346 040722
 04347 141040
 04350 143715
 04351 040705
 04352 150043
 04353 151317
 04354 120134
 04355 151015
 04356 147722
 04357 051723
 04360 000011
11                      ;ADDRESS ERROR
12              MSG29:  .TXTE |ADDRESS ERROR           |
 04361 042101
 04362 151124
 04363 051705
```

```
0009   .MAIN
 04304 120123
 04305 151305
 04306 147722
 04307 004722
 04370 000011
01                          ;PERM ADDRESS ERROR
02                  MSG30:  .TXTE |PERM ADDRESS ERROR                |
 04371 142520
 04372 040722
 04373 040040
 04374 042104
 04375 142722
 04376 051523
 04377 142040
 04400 151322
 04401 151317
 04402 004411
 04403 000000
03                          ;DATA ERRORS
04                  MSG31:  .TXTE |DATA ERRORS                       |
 04404 040504
 04405 040724
 04406 142040
 04407 151322
 04410 151317
 04411 004523
 04412 000011
05                          ;READ
06                  MSG32:  .TXTE |READ      |
 04413 142722
 04414 042101
 04415 000011
07                          ;WRITE
00                  MSG33:  .TXTE |WRITE     |
 04416 151327
 04417 152311
 04420 004705
 04421 000000
09                          ;MEMORY TOO SMALL
10                  MSG39:  .TXTE |MEMORY TOO SMALL FOR |
 04422 142515
 04423 147515
 04424 004722
 04425 150040
 04426 142717
 04427 051503
 04430 040515
 04431 140514
 04432 140040
 04433 151317
 04434 000240
11                          ;SECTORS
12                  MSG40:  .TXTE |SECTORS|
 04435 140523
 04436 142523
 04437 151317
 04440 000123
13                          ;PASS
14                  MSG41:  .TXTE |PASS|
```

```
  0070   .MAIN
   04441 040520
   04442 051523
   04443 000000
01                          ; -?-
02                MSG43:    .TXTE | -?-|
   04444 026640
   04445 026477
   04446 000000
03                          ;CYL-
04                MSG44:    .TXTE |CYL-|
   04447 054703
   04450 026714
   04451 000000
05                          ;HEAD-
06                MSG45:    .TXTE |HEAD-|
   04452 142510
   04453 042101
   04454 000055
07                          ;SECT-
08                MSG46:    .TXTE |SECT-|
   04455 142523
   04456 152303
   04457 000055
09                          ;#SECT-
10                MSG47:    .TXTE |#SECT-|
   04460 051643
   04461 141705
   04462 026724
   04463 000000
11                          ;GOOD
12                MSG51:    .TXTE |GOOD |
   04464 147507
   04465 042317
   04466 000240
13                          ;BAD
14                MSG52:    .TXTE |BAD |
   04467 040502
   04470 120104
   04471 000000
15                          ;"DOC"
16                MSG53:    .TXTE |"DOC"|
   04472 042042
   04473 141717
   04474 000042
17                          ;"DOS"
18                MSG54:    .TXTE |"DOS"|
   04475 042042
   04476 141723
   04477 000042
19                          ;LOAD ERROR
20                MSG55:    .TXTE | LOAD ERROR|
   04500 154117
   04501 042717
   04502 121104
   04503 151522
   04504 162722
   04505 000040
21                          ;ERRORS
22                MSG56:    .TXTE |ERRORS|
```

```
0071  .MAIN
  04506 151305
  04507 147722
  04510 051722
  04511 000000
                         ;ENDING MEM ADDR ERROR
              MSG57:     .TXTE |ENDING MEM ADDR ERROR|
  04512 047305
  04513 144504
  04514 043510
  04515 040540
  04516 040705
  04517 040640
  04520 042104
  04521 120322
  04522 151305
  04523 147722
  04524 000322
                         ;TIMEOUT
              MSG59:     .TXTE |TIMEOUT|
  04525 144724
  04526 142515
  04527 052717
  04530 000324
                         ;PERM DATA ERRORS
              MSG61:     .TXTE |PERM DATA ERRORS          |
  04531 142520
  04532 046722
  04533 042240
  04534 152101
  04535 120101
  04536 151305
  04537 147722
  04540 051722
  04541 004411
  04542 000000
                         ;MISC ERRORS
              MSG62:     .TXTE |MISC ERRORS               |
  04543 144515
  04544 141523
  04545 142040
  04546 151322
  04547 151517
  04550 004520
  04551 000011
                         ;TRY AGAIN
              MSG63:     .TXTE |TRY AGAIN|
  04552 151024
  04553 120131
  04554 040501
  04555 144511
  04556 000116
                         ;CHECK WORD ERROR
              MSG64:     .TXTE |CHECK WORD ERROR|
  04557 [illegible]
  04560 [illegible]
  04561 [illegible]
  04562 147727
  04563 042522
  04564 142040
```

```
0072  .MAIN
  04565 151322
  04566 151317
  04567 000000
                                  ;CHECK WORD & DATA ERROR
                         MSG65:   .TXTE |CHECK WORD & DATA ERRORS |
  04570 044303
  04571 141705
  04572 120113
  04573 147727
  04574 042322
  04575 123240
  04576 042240
  04577 152101
  04600 120101
  04601 151303
  04602 147722
  04603 051722
  04604 000011
                                  ;PERM CHK WD & DATA ERRS
                         MSG66:   .TXTE |PERM CHK WD & DATA ERRS   |
  04605 142520
  04606 046722
  04607 141640
  04610 045510
  04611 153640
  04612 120104
  04613 120246
  04614 040504
  04615 040724
  04616 142640
  04617 151322
  04620 004523
  04621 000000
                                  ;ENDING DISK ADDR ERROR
                         MSG67:   .TXTE |ENDING DISK ADDR ERROR|
  04622 047305
  04623 144504
  04624 043516
  04625 042240
  04626 051711
  04627 120113
  04630 042101
  04631 151104
  04632 142040
  04633 [illegible]
  04634 [illegible]
  04635 [illegible]
                                  ;INTERCHANGE DISK
                         MSG68:   .TXTE |INTERCHANGE DISK|
  04636 [illegible]
  04637 [illegible]
  04640 [illegible]
  04641 [illegible]
  04642 [illegible]
  04643 [illegible]
  04644 [illegible]
  04645 045023
  04646 000000
                                  ;TYPE THE NUMBER OF DISK SURFACES
```

```
  0073  .MAIN
01                      MSG69:  .TXTE |TYPE THE NUMBER OF DISK SURFACES |
    04647 054724
    04650 140520
    04651 152040
    04652 142510
    04653 047240
    04654 040525
    04655 142502
    04656 120322
    04657 143317
    04660 042240
    04661 051711
    04662 120113
    04663 052523
    04664 143322
    04665 141501
    04666 051705
    04667 000240
02                              |TESTING UNIT
03                      MSG70:  .TXTE |TESTING UNIT  |
    04670 142724
    04671 152123
    04672 047311
    04673 120107
    04674 047125
    04675 152311
    04676 120240
    04677 000000
04                              |SET SW4 FOR INTERCHANGE, SW5 FOR READ ONLY
05                      MSG71:  .TXTE |SET SW4 FOR INTERCHANGE, SW5 FOR READ ON
    04700 142523
    04701 120324
    04702 150523
    04703 120254
    04704 147700
    04705 120322
    04706 047311
    04707 142724
    04710 141722
    04711 040510
    04712 040510
    04713 120000
    04714 051040
    04715 000727
    04716 140515
    04717 151517
    04720 151524
    04721 040720
    04722 120104
    04723 047317
    04724 000714
    04725 000000
06                              |STRIKE ANY KEY TO CONTINUE
07                      MSG72:  .TXTE |STRIKE ANY KEY TO CONTINUE|
    04726 123122
    04727 140522
    04730 142510
    04731 040040
    04732 054516
```

```
0074   .MAIN
 04733 045040
 04734 054705
 04735 152240
 04736 120317
 04737 147703
 04740 152116
 04741 047311
 04742 142525
 04743 000000
```

```
^ 0075   .MAIN
01
02          000072 UDUFF:   .BLK 50.
03          000072 CDUFF:   .BLK 50.
04          000020 VAR:     .BLK 16.
05  05150  000000 PRGEND:  0
06
07
08                          .END
```