T68

```
0001   .MAIN

                                        ;RCSL:    44-RT 1661
                                        ;AUTHOR:  FLEMMING KAM
                                        ;EDITED:  78.02.07

;               RC 3600

;           DCC 701 DIAGNOSTIC.

;S-BINARY TAPE: RCSL 44-RT 1662 (SELFSTART AUTOLOAD HEAD).
;BINARY CARD:   RCSL 44-RT 1663

;KEYWORDS:      RC 3652, RC 3688, F22.
;              DCD 701, DCC 701, DCA 701.

;ABSTRACT:      THIS PROGRAM IS A HARDWARE DIAGNOSTIC FOR
;              THE RC 3688 2,4MB DISC CARTRIDGE CONTROLLER
;              AND ADAPTER LOGIC. IT IS ASSUMED THAT THE
;              DISC TERMINAL IS FUNCTION PROPERLY.
;       NOTE:  THE DIAGNOSTIC IS ABLE TO USE THE EXTENDED
;              MEMORY AS BUFFER.
```

```
↑ 0002  .MAIN
                ;DISC CARTRIDGE CONTROL DIAGNOSTIC.

               ;******  THE ABILITY OF ACCESING THE EXTENDED MEMORY
               ;        INCLUDED 78.03.07.
               ;
               ; 1. ABSTRACT
               ;       THIS PROGRAM IS A HARDWARE DIAGNOSTIC FOR THE RC 3652
               ;       MOVING HEAD DISC CRONTROLLER AND ADAPTER LOGIC.
               ;       IT IS ASSUMED THAT THE DISC TERMINAL IS
               ;       FUNCTIONING PROPERLY.
               ;*NOTE  THE PROGRAM IS ABLE TO USE THE EXTENDED
               ;       MEMORY ( > 32K) AS BUFFER.

               ; 2. REQUIREMENTS
               ;       1. RC 3600 FAMILY PROCESSOR
               ;     - 2. MINIMUM OF 4K READ/WRITE MEMORY
               ;       3. RC 3688 DISC CARTRIDGE CONTROLLER, DCC 701.
               ;       4. F22 DISC CARTRIDGE ADAPTER, DCA 701
               ;       5. 1 TO 4 DISC TERMINALS, RC 3652.
               ;       6. TELETYPE AND CONTROL

               ; 3. OPERATING PROCEDURE.
               ;       1. LOAD USING THE AUTOLOAD, BINLOADER OR
               ;          RC 3600 HARDWARE TEST SYSTEM.
               ;       2. STARTING ADDRESSES:
               ;
               ;               400 - DIAGNOSTIC, NORM MODE, (MEM <=32K).
               ;
               ;                NOTE SWITCHES: CPU 68/128K, DCC NORMAL,
               ;                     BEFORE START IS IT POSSIBLE TO SET THE
               ;                     READ/WRITE BUFFER START, LOC 374(>=DEF7220)
               ;                     READ BUFFER START, LOC 375 (>=DEFAULT 7620)

               ;               401 - DIAGNOSTIC, EXT MODE (MEM > 32K).

               ;                NOTE SWITCHES: CPU 128K, DCC EXTENDED.
               ;                     BEFORE START IS IT POSSIBLE TO SET THE
               ;                     READ/WRITE BUFFER START, LOC 376(DEF107220)
               ;                     READ BUFFER START, LOC 377 (DEFAULT 107620)
               ;
               ;               402 - RANDOM SEEK EXCERCISER, NORM MODE.

               ;               403 - RANDOM SEEK EXCERCISER, EXT MODE.
               ;
               ;       3. THE PROGRAM PRINTS "PASS" FOLLOWING EACH
               ;          COMPLETE PASS THROUGH THE TEST. RANDOM
               ;          SEEK EXERCISER PERFORMS 5000 SEEKS
               ;          PER "PASS" MESSAGE.

               ;       4. SWITCH SETTINGS.
               ;               SW1= FROM ERROR, GO TO NEXT TEST.
               ;               SW2= INHIBIT TTY OUTPUT.
               ;               SW3= PRINT FAILURE RATE.
               ;               SW5= OUTPUT TO LPT.
               ;               SW6-7= UNIT # FOR RECAL DURING SCOPE LOOP
               ;               SW8= RECALIBRATE DURING SCOPE LOOP.
               ;               SW9= 1 SEC DELAY IN 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 SW3 TO CAUSE THE ERROR RATE (0-100%) TO BE
               ;       PRINTED. SET SW1 TO EXIT FROM THE SCOPE LOOP
               ;       AND PROCEED TO THE NEXT TEST.
```

```
; 5.MODIFYCATIONS.
;       THIS PROGRAM IS A MODIFYED VERSION OF THE DGC
;       MOVING HEAD DISC DIAGNOSTIC, REV 10. IT IS CHANGED
;       IN FOLLOWING MANNER:
;
;    A. A PIECE OF CODE FROM LOC 300 TO 467 IS INSERTED.
;
;    B. A TABLE FROM LOC 7110 TO 7512 IS APPENDED.
;
;    C. FOLLOWING INSTRUCTIONS ARE PERMANENT ALTERED:
;
;    PAGE/LINE  LOC     FROM            TO
;
;       4/10    6       JMP @IRDSK      JMP ILLEG (000310)
;     - 7/13    200     JMP BGNADR      JMP ILLEG (000310)
;       50/43   3634    LDA 1,@E8.2     JSR GETDA (004325)
;       51/44   3707    LDA 1,@E10.2    JSR GETDA (004325)
;       53/17   3776    LDA 1,@E12.1    JSR GETDA    "
;       55/24   4103    LDA 1,@E14.1    JSR GETDA    "
;       58/17   4177    LDA 1,@E16.1    JSR GETDA    "
;       60/26   4321    LDA 1,@E18.1    JSR GETDA    "
;       64/34   4546    LDA 1,@BUFF     JSR GETDA    "
;       64/49   4565    LDA 1,@BUFF     JSR GETDA    "
;       83/23   5640    JMP .LOP1+1     JMP .LOP1 (000434)
;       84/21   5674    IORST           JSR RESET (004321)
;
;    D. BEFORE START OF THE DIAGNOSTIC FOLLOWING MODIFYCATINOS
;       ARE PERFORMED BY THE PROGRAM:
;
;         NORMAL MODE.
;
;       ALL THE OCCURRENCE PRGEN, PRGEN+400 AND THE CONTENTS OF
;       OF BUFF (LOC 153) ARE ALTERED FROM RESPECTIVELY 6630,7230
;       AND 6630 TI THE CONTENTS OF RESPECTIVELY
;       LOC 374, 375 AND 374.
;
;         EXTENDED MODE.
;
;       ALL THE OCCURRENCE OF PRGEN, PRGEN+400 AND THE
;       CONTENTS OF BUFF (LOC 153) ARE ALTERED FROM
;       RESPECTIVELY 6630,7230 AND 6630 TO THE CONTENTS
;       OF RESPECTIVELY LOC 376, 377 AND 376.
```

```
 0001 .MAIN MACRO REV 03.00              12:01:06 07/16/76
01
02
03
04
05
06
07              ;*************************************************************
08              ;
09              ; NAME: MHOD.SR                          PART NUMBER: 094-000211
10              ;
11              ;
12              ; DESCRIPTION: MOVING HEAD DISK CONTROL DIAGNOSTIC
13              ;
14              ;
15              ; REVISION HISTORY:
16              ;
17              ;       REV.            DATE
18              ;
19              ;       00              10/20/71
20              ;       01              06/13/72
21              ;       02              11/20/72
22              ;       03              07/03/73
23              ;       04              08/24/73
24              ;       05              04/20/74
25              ;       06              XX/XX/XX
26              ;       07              XX/XX/XX
27              ;       08              12/12/75
28              ;       09              04/23/76
29              ;       10              07/16/76
30              ;
31              ; COPYRIGHT (C) DATA GENERAL CORPORATION, 1971, 1972, 1973, 1974
32              ; 1975,1976 ALL RIGHTS RESERVED.
33              ;*************************************************************
```

```
 0002 .MAIN
01
02                      ;MOVING HEAD DISK CONTROL DIAGNOSTIC
03
04              ;****** AUTO-RUN AUTO LOAD MODIFIED 3/7/72
05              ;END OF CYLINDER TEST INCLUDED 06/28/73
06              ;100 MS TIMES FOR LOOPS 08/07/73
07
08              ; 1. ABSTRACT
09              ;       THIS PROGRAM IS A HARDWARE DIAGNOSTIC FOR THE
10              ;       4046 MOVING HEAD DISK CONTROLLER AND ADAPTER
11              ;       LOGIC.  IT IS ASSUMED THAT THE DISK TERMINAL
12              ;       IS FUNCTIONING PROPERLY.
13              ;
14              ; 2. REQUIREMENTS
15              ;       1. NOVA (EXCEPT MICRO) OR ECLIPSE FAMILY CENTRAL PROCESS
16              ;       2. MINIMUM OF 4K READ/WRITE MEMORY
17              ;       3. 4046 MOVING HEAD DISK CONTROL
18              ;       4. 4047,4048,4049 OR 4057
19              ;          DISK ADAPTER
20              ;       5. 1 TO 4 DISK TERMINALS
21              ;       6. TELETYPE AND CONTROL
22              ;
23              ; 3. OPERATING PROCEDURE
24              ;       1. LOAD USING THE BINARY LOADER OR DIAGNOSTIC
25              ;          OPERATING SYSTEM.
26              ;       2. STARTING ADDRESSES:
27              ;             6= RANDOM SEEK EXCERCISER
28              ;            200= START DIAGNOSTIC
29              ;       3. THE PROGRAM PRINTS "PASS" FOLLOWING EACH
30              ;          COMPLETE PASS THROUGH THE TESTS. RANDOM
31              ;          SEEK EXERCISER PERFORMS 5000 SEEKS
32              ;          PER "PASS" MESSAGE.
33              ;       4. SWITCH SETTINGS:
34              ;             SW1= FROM ERROR, GO TO NEXT TEST.
35              ;             SW2= INHIBIT TTY OUTPUT.
36              ;             SW3= PRINT FAILURE RATE.
37              ;             SW5= OUTPUT TO LPT
38              ;             SW6-7= UNIT # FOR RECAL DURING SCOPE LOOP
39              ;             SW8= RECALIBRATE DURING SCOPE LOOP
40              ;             SW9= 1 SEC DELAY IN SCOPE LOOP
41              ;
42              ; 4. ERRORS
43              ;       WHEN AN ERROR IS DETECTED THE PROGRAM HALTS.
44              ;       (AC3) POINTS TO THE LOCATION FOLLOWING THE
45              ;       ERROR HALT CALL "EHALT".  CONSULT THE COMMENTS
46              ;       AREA OF THE DIAGNOSTIC PROGRAM LISTING FOR
47              ;       CLUES AND POSSIBLE CAUSES OF THE FAILURE.
48              ;       PUSHING CONTINUE WILL CAUSE THE PROGRAM TO
49              ;       PRINT THE (AC3) AND ENTER A SCOPE LOOP.
50              ;       SET SW3 TO CAUSE THE ERROR RATE (0-163X) TO BE
51              ;       PRINTED.  SET SW1 TO EXIT FROM THE SCOPE LOOP
52              ;       AND PROCEED TO THE NEXT TEST.
```

```
10003 .MAIN
01
02                ;       SOME SCOPE LOOPS WILL REQUIRE A RECALIBRATE
03                ;       TO INITIALIZE THE DISK DRIVE FOLLOWING A FAILURE.
04                ;       SET SWITCH 8 TO INTRODUCE THE RECALIBRATE.  THE
05                ;       UNIT TO BE RECALIBRATED MUST BE SET INTO SWITCHES
06                ;       6 AND 7.
07                ;
08                ;       TESTS THAT PERFORM A RECALIBRATE HAVE A 2 SEC.
09                ;       DELAY BUILT INTO THE SCOPE LOOP AS PROTECTION
10                ;       FOR THE DISK DRIVE ELECTRONICS.  SET SWITCH 9
11                ;       TO INTRODUCE AN ADDITIONAL 1 SECOND DELAY DURING
12                ;       THE SCOPE LOOP.
13                ;
14                ;       IN GENERAL EACH SUCCESSIVE TEST ASSUMES ALL
15                ;       PREVIOUS TESTS WORK.  BYPASSING ERRORS
16                ;       CAN RESULT IN CONFUSING SITUATIONS
17                ;       IN THE SETUP OF MORE COMPLEX TESTS.
18                ;
19                ; 5. DISK PACKS
20                ;       ONLY USE DISK PACKS FORMATTED BY THE DGC DISK
21                ;       PACK FORMATTER PROGRAM.  THE DIAGNOSTIC PROGRAM
22                ;       WILL WRITE OVER MOST OF THE DISK SURFACE.
23                ;       THE FORMAT MODE IS NOT CHECKED.
```

```
10004 .MAIN
01        000000            .LOC 0          ;DO NOT INSERT. LOC 0
02                                          ;CONTAINS DIRT POINTER
03 00000 006666             DIRT           ;DIRT POINTER
04
05 00001 005564             IRET           ;INTERRUPT RETURN
06 00002 000200             JMP 200
07 00003 000200             JMP 200
08 00004 000200             JMP 200
09 00005 000200             JMP 200
10 00006 002007             JMP @IRDSK     ;TO RANDOM SEEK EXC.
11 00007 005506 IRDSK:      RANSK
12 00010 000002 C2:         2
13 00011 000000 EGGS:       0              ;AUTO RUN SWITCH
14 00012 000000             0              ;DEVICE CODE
15 00013 000000             0              ;NOT USED
16 00014 000000             0              ;# PASSES THIS RUN
17 00015 000000             0              ;DIAGNOSTIC SYSTEM RETURN ADR.
18 00016 000000             0              ;SWITCH REGISTER
19
20        000045            .LOC 45         ;DO NOT INSERT. LOC 45
21                                          ;CONTAINS EGGS POINTER
22 00045 000011             EGGS
23
24 00046 000000 .TU:        0
25 00047 040000             40000
26 00050 100000             100000
27 00051 140000             140000
28 00052 000001 UNTBIT:     1
29 00053 000002 KB14:       2
30 00054 000004 KB13:       4
31 00055 000010 KB12:       10
32 00056 000020 KB11:       20
33 00057 000040 KB10:       40
34 00060 000100 KB9:        100
35 00061 000200 KB8:        200
36 00062 000400 KB7:        400
37 00063 001000 KB6:        1000
38 00064 002000 KB5:        2000
39 00065 004000 KB4:        4000
40 00066 010000 KB3:        10000
41 00067 020000 KB2:        20000
42        000047 KB1=.TU+1
43
44 00070 177775 ZB14:       177775
45 00071 177773 ZB13:       177773
46 00072 177767 ZB12:       177767
47 00073 177757 ZB11:       177757
48 00074 177737 ZB10:       177737
49 00075 177677 ZB9:        177677
50 00076 177577 ZB8:        177577
51 00077 177377 ZB7:        177377
52 00100 176777 ZB6:        176777
53 00101 175777 ZB5:        175777
54 00102 173777 ZB4:        173777
55 00103 167777 ZB3:        167777
56 00104 157777 ZB2:        157777
57 00105 137777 ZB1:        137777
58
59 00106 006242 TRCL:       RECL0
60 00107 006243             RECL1
```

```
 0005 .MAIN
01 00110 006244          RECL2
02 00111 006245          RECL3
```

```
0005 .MAIN
01
02 00112 000003 C3:      3
03 00113 000007 C7:      7
04 00114 000017 C17:     17
05 00115 000037 C37:     37
06 00116 000077 C77:     77
07 00117 000177 C177:    177
08 00120 000377 C377:    377
09 00121 000777 C777:    777
10 00122 001777 C1777:   1777
11 00123 003777 C3777:   3777
12 00124 007777 C7777:   7777
13 00125 017777 C017:    17777
14 00126 037777 C037:    37777
15
16 00127 010421 BIT1:    010421
17 00130 021042 BIT2:    021042
18 00131 042104 BIT4:    042104
19 00132 104210 BIT8:    104210
20
21 00133 052525 C2525:   052525
22 00134 125252 C5252:   125252
23 00135 123456 RANDOM:  123456
24 00136 123456 RELRAN:  123456
25 00137 000000 TEMP:    0
26 00140 000000 TIME:    0
27 00141 000000 TIME1:   0
28 00142 056000 KDOB:    DDB 1,0
29 00143 000000 DTYPE:   0                ;0=CART, 1=2311, 15=2314
30 00144 000001 NDSKS:   000001           ;15=UNIT 0, 14=1, 13=2, 12=3
31 00145 000000 TESTU:   0                ; 01 = UNIT #
32 00146 000000 UNUM:    0                ; 14-15 = UNIT #
33 00147 000033 CDSK:    33
34 00150 000000 CYL:     0
35 00151 000000 HEAD:    0
36 00152 000000 SECT:    0
37 00153 006630 BUFF:    PRGEND
38 00154 160037 MSK1:    160037
39 00155 060033 DPIO:    060033
40 00156 177700 MSK2:    177700
41 00157 000000 ITRCNT:  0
42 00160 005422 STALL:   .STL
```

```
!0007 .MAIN
01
02 00161 000004 C4:     4
03 00162 000005 C5:     5
04 00163 000006 C6:     6
05 00164 000011 C11:    11
06 00165 000012 C12:    12
07 00166 000015 C15:    15
08 00167 000016 C16:    16
09 00170 000030 C30:    30
10 00171 000033 C33:    33
11 00172 000036 C36:    36
12       000200         .LOC 200
13 00200 000202 DTOSB:  JMP BGNADR      !DTOS STARTS HERE
14 00201 000000 LOOPR:  0
15 00202 000230 BGNADR: BEGIN
16 00203 000060 C60:    60
17 00204 000063 C63:    63
18 00205 000070 C70:    70
19 00206 000120 C120:   120
20 00207 000137 C137:   137
21 00210 000157 C157:   157
22 00211 000277 C277:   277
23 00212 000312 C312:   312
24 00213 000317 C317:   317
25 00214 000402 C402:   402
26 00215 000420 C420:   420
27 00216 177400 C1774:  177400
28
29
30 00217 177372 ICMSK:  177372
31 00220 000073 C73:    73
32 00221 177400 M400:   -400
33 00222 001400 C1400:  1400
34 00223 003000 C3000:  3000
35 00224 002400 C2400:  2400
36 00225 074000 C74K:   74000
37       000051 C140K=.TU+3
38 00226 174000 C174K:  174000
39       000003 C1000=KB6
40
41 00227 005413 DEVCK:  DEVC
42 00230 006261 BEGIN:  MESSAGE
43 00231 005441         MSG0
44 00232 063077         HALT
45 00233 005227         JSR @DEVCK
46 00234 020171         LDA 0,C33
47 00235 100414         SUB# 0,1,SZR
48 00236 002256         JMP @IS73
49 00237 002255         JMP @IS33
```

```
!0008 .MAIN
01 00240 005444 I.WAIT: .WAIT
02 00241 005431 I.SSEK: .SSEK
03 00242 005463 I.RC0:  .RCL0
04 00243 005465 I.RC1:  .RCL1
05 00244 005467 I.RC2:  .RCL2
06 00245 005470 I.RC3:  .RCL3
07 00246 005543 I.IWT:  .IWT
08 00247 005566 I.ADSK: .ADSK
09 00250 005515 ISET:   .SET
10 00251 005020 I.SETP: .SETP
11 00252 005023 I.STUP: .SETUP
12 00253 005774 I.EHA:  .EHALT
13 00254 005654 I.LOO:  .LOOP
14 00255 005371 IS33:   .S33
15 00256 005372 IS73:   .S73
16 00257 005374 I.S:    .SK
17 00260 006250 ICRLF:  CRLF
18 00261 006105 IMESS:  MESS
19 00262 005236 I.INI:  .INI
20 00263 005771 I.EHI:  .EHI
21 00264 005501 I.DO:   .DORW
22 00265 005012 IRAN:   RAN
23 00266 005071 IGEN:   .GEN
24 00267 005131 IREAD:  .READ
25 00270 005104 IWRT:   .WRITE
26 00271 005247 ICHK:   .CHECK
27 00272 005147 IDOS:   .DOSEK
28 00273 005552 I.LD:   .LUPD
29 00274 000000 SVTIM:  0
30 00275 006334 ITIN:   TIND
31 00276 006126 PEX:    PEXIT+1
32 00277 000000 SW:     0
33
34       006252 .DUSR   SETUP=JSR @I,STUP
35       006251 .DUSR   SETP1=JSR @I,SETP
36       006253 .DUSR   EHALT=JSR @I,EHA
37       006254 .DUSR   LOOP=JSR @I,LOO
38       000033 .DUSR   DSKP=33
39       006263 .DUSR   EHLT=JSR @I,EHI
40       006273 .DUSR   LOOPD=JSR @I,LD
41
42       006261 MESSAGE=JSR @IMESS
43       006260 PCRLF=JSR @ICRLF
44       006240 WAIT=JSR @I,WAIT
45       006241 SSEEK=JSR @I,SSEK
46       006242 RECL0=JSR @I,RC0
47       006243 RECL1=JSR @I,RC1
48       006244 RECL2=JSR @I,RC2
49       006245 RECL3=JSR @I,RC3
50       006246 ITRWT=JSR @I,IWT
51       006247 GADSK=JSR @I,ADSK
52       006257 SEEK=JSR @I,S
53       006262 INIT=JSR @I,INI
54       006266 GENDAT=JSR @IGEN
55       006267 READ=JSR @IREAD
56       006270 WRITE=JSR @IWRT
57       006271 CHECK=JSR @ICHK
58       006272 DOSEK=JSR @IDOS
59       006264 DORW=JSR @I,DO
60
```

```
0001  .MAIN

      000300            .LOC    300
00300 100001 C.1MM:     100001
00301 002614 ADDR1:     2614            ;ADDRESS
00302 002632 ADDR2:     2632            ;ADDRESS
00303 152520 ND1:       SUBZL   2,2     ;NORM INSTR.
00304 152620 ND2:       SUBZR   2,2     ; "     "
00305 030300 XD1:       LDA     2,300   ;EXT    "
00306 152400 XD2:       SUB     2,2     ; "     "

      000310            .LOC    310
00310 063077 ILLEG:     HALT            ;ILLEGAL START ADDRESS.
00311 000310            JMP     .-1

      000320            .LOC    320
00320 000000 WK:        0               ;LINK
00321 054320 RESET:     STA     3,WK
00322 062677            IORSI
00323 076701            DICP    3,1     ;SET MEM EXT FLAG.
00324 002320            JMP     @WK     ;RETURN

      000325            .LOC    325
00325 054320 GETDA:     STA     3,WK    ;STORE LINK
00326 034153            LDA     3,153   ;GET BUFFER START.
00327 025400            LDA     1,0,3   ;GET 1. WORD OF DATA BUFFER.
00330 002320            JMP     @WK     ;RETURN

      000374            .LOC    374
00374 007220 NWRITE:    7220            ;NORM READ/WRITE BUFFER START, > 7220
00375 007620 NREAD:     7620            ;  "  READ    "     "      "   > 7220+4
00376 107220 XWRITE:    107220          ;EXT READ/WRITE BUFFER, > 7220
00377 107620 XREAD:     107620          ; "  READ  "    "       > 7220+400


00400 000404            JMP     SA400   ;DIAGNOSTIC - NORM MODE.
00401 000407            JMP     SA401   ;DIAGNOSTIC - EXT MODE.
00402 000412            JMP     SA402   ;SEEK EXCERCISER - NORM MODE.
00403 000415            JMP     SA403   ;SEEK EXCERCISER - EXT MODE.

00404 004431 SA400:     JSR     NORM    ;SET INSTR FOR NORM MODE.
 0405 030461            LDA     2,BGNADR
00406 050456            STA     2,RETURN
00407 000437            JMP     COMM    ;START PROC.

00410 004414 SA401:     JSR     EXT     ;SET INSTR FOR EXT MODE.
00411 030455            LDA     2,BGNADR
00412 050452            STA     2,RETURN
00413 000433            JMP     COMM    ;START PROC.

00414 004421 SA402:     JSR     NORM    ;SET INSTR. FOR NORM MODE.
00415 030452            LDA     2,IRDSK
00416 050446            STA     2,RETURN
00417 000427            JMP     COMM    ;START PROC

00420 004404 SA403:     JSR     EXT     ;SET INSTR FOR EXT MODE.
00421 030446            LDA     2,IRDSK
00422 050442            STA     2,RETURN
00423 000423            JMP     COMM    ;START PROC.
```

```
00424 030301 EXT:    LDA     2,ADDR1 ;SET INSTR FOR EXT MODE.
00425 020305         LDA     0,XD1
00426 041000         STA     0,0,2
00427 030302         LDA     2,ADDR2
00430 020306         LDA     0,XD2
00431 041000         STA     0,0,2
00432 020376         LDA     0,XWRITE
00433 024377         LDA     1,XREAD
00434 001400         JMP     0,3     ;RETURN


00435 030301 NORM:   LDA     2,ADDR1 ;ROUTINE TO SET INSTR FOR
00436 020303         LDA     0,ND1   ;NORM MODE.
00437 041000         STA     0,0,2
00440 030302         LDA     2,ADDR2
00441 020304         LDA     0,ND2
00442 041000         STA     0,0,2
00443 020374         LDA     0,NWRITE
00444 024375         LDA     1,NREAD
00445 001400         JMP     0,3     ;RETURN.

00446 030417 COMM:   LDA     2,TABST ;CODE TO SET NORM/EXT BUFFER
00447 035000         LDA     3,0,2   ;ADDRESSES IN PROG.
00450 175005         MOV     3,3,SNR ;WRITE BUFFERS.
00451 000404         JMP     COM2
00452 041400         STA     0,0,3
00453 151400         INC     2,2
00454 000773         JMP     COMM+1

00455 151400 COM2:   INC     2,2     ;READ BUFFERS.
00456 035000         LDA     3,0,2
00457 175005         MOV     3,3,SNR
00460 000403         JMP     COM3
00461 045400         STA     1,0,3
00462 000773         JMP     COM2

00463 002401 COM3:   JMP     @RETURN ;FINIS - JMP TO DIAG/SEEK.

00464 000000 RETURN: 0
00465 007110 TABS1:  7110
00466 000230 BCNADR: 230
00467 004506 IRDSK:  4506
```

```
 0000 .MAIN
01          000500          .LOC 500
02
03 000500 002441            JMP @.+1
04 000501 002240            DTOSB
05 000502 006262 START:     INIT            ;DO IT NOW.
06
07 000503 020144 A0:        LDA 0,NDSKS     ;UNIT 0 SPECIFIED ?
08 000504 101202            MOVR 0,0,SZC
09 000505 006242            RECL0           ;YES, RECALIBRATE IT
10 000506 062677            IORST
11
12 000507 020144            LDA 0,NDSKS     ;UNIT 1 SPECIFIED ?
13 000510 024053            LDA 1,KB14
14 000511 123404            AND 1,0,SZR
15 000512 006243            RECL1           ;YES RECALIBRATE IT
16 000513 062677            IORST
17
18 000514 020144            LDA 0,NDSKS     ;UNIT 2 SPECIFIED ?
19 000515 024054            LDA 1,KB13
20 000516 123404            AND 1,0,SZR
21 000517 006244            RECL2           ;YES, RECALIBRATE IT
22 000520 062677            IORST
23
24 000521 020144            LDA 0,NDSKS     ;UNIT 3 SPECIFIED ?
25 000522 024055            LDA 1,KB12
26 000523 123404            AND 1,0,SZR
27 000524 006245            RECL3           ;YES, RECALIBRATE IT
```

```
;0010 .MAIN
01
02 000525 006252 A1:        SETUP           ;CHECK SELD BUS LINE
03 000526 063700            SKPDZ 0         ;SKIP IF LINE HIGH
04 000527 006253            EHALT           ;DSKP HAS SELD GROUNDED.
05 000530 006254            LOOP
06
07 000531 006252 A2:        SETUP           ;CHECK SELB BUS LINE
08 000532 063500            SKPBZ 0         ;SKIP IF LINE HIGH
09 000533 006253            EHALT           ;DSKP HAS SELB GROUNDED
10 000534 006254            LOOP
11
12 000535 006252 A3:        SETUP           ;CHECK DISK PACK BUSY
13 000536 063533            SKPBZ DSKP      ;SKIP IF BUSY 0
14 000537 006253            EHALT           ;"DP BUSY" STUCK ON
15 000540 006254            LOOP
16
17 000541 006252 A4:        SETUP           ;CHECK DISK PACK DONE
18 000542 063733            SKPDZ DSKP      ;SKIP IF DONE 0
19 000543 006253            EHALT           ;"DP DONE" STUCK ON
20 000544 006254            LOOP
01
22 000545 006252 A5:        SETUP           ;CHECK I/O DATA LINES
23 000546 060400            DIA 0,0         ;DIA TO DEVICE 0
24 000547 101004            MOV 0,0,SZR
25 000550 006253            EHALT           ;GROUNDED DATA LINE(S)
26 000551 006254            LOOP
27
28 000552 006252 A6:        SETUP           ;CHECK CA REGISTER FOR
29 000553 061433            DIB 0,DSKP      ;ZEROS AFTER "RESET"
30 000554 101004            MOV 0,0,SZR     ;POSSIBLE FAILURE OF "RESET"
31 000555 006253            EHALT           ;OR TKF REGISTER.
32 000556 006254            LOOP
33
34 000557 006252 A7:        SETUP           ;CHECK DISK ADDRESS REG.
35 000560 062433            DIC 0,DSKP      ;FOR ZEROS FOLLOWING IORST
36 000561 101004            MOV 0,0,SZR     ;FAILING REGISTER IC OR
37 000562 006253            EHALT           ;POSSIBLE FAILURE OF "RESET"
38 000563 006254            LOOP            ;OR "RESET 5".
39
40 000564 006252 A8:        SETUP           ;TRY TO LOAD CA WITH
41 000565 102000            ADC 0,0         ;ALL ONES
42 000566 062033            DOB 0,DSKP      ;LOAD CA REGISTER
43 000567 065433            DIB 1,DSKP      ;READ IT BACK
44 000570 122434            SUBZR 1,0,SZR   ;CHECK "DP DATOB",
45 000571 006253            EHALT           ;"DPDATIB", CA REGISTER,
46 000572 006254            LOOP            ;AND DATA PATH THRU MUXES.
```

```
10011 .MAIN
#1
#2 00573 006252 A9:     SETUP           ;SEE IF DOB LOADS
#3 00574 102200         ADC 0,0         ;DISK ADDRESS REGISTER
#4 00575 062033         DOB 0,DSKP      ;LOAD CA REG.
#5 00576 066433         DIC 1,DSKP      ;READ DISK ADDR REG.
#6 00577 125004         MOV 1,1,SZR     ;IT SHOULD STILL BE ALL #1S
#7 00600 006253         EHALT
#8 00601 006254         LOOP
#9
10 00602 006252 A10:    SETUP           ;SEE IF DOC LOADS THE
11 00603 102200         ADC 0,0         ;CA REGISTER
12 00604 063033         DOC 0,DSKP      ;LOAD DISK ADDR REG
13 00605 065433         DIB 1,DSKP      ;READ CA REGISTER
14 00606 125004         MOV 1,1,SZR     ;CA REGISTER SHOULD
15 00607 006253         EHALT           ;REMAIN ALL ZERO
16 00610 006254         LOOP
17
18 00611 006252 A11:    SETUP           ;SEE IF THE DISK ADDRESS
19 00612 102200         ADC 0,0         ;REGISTER EXISTS
20 00613 063033         DOC 0,DSKP      ;LOAD IT WITH ALL 1'S
21 00614 066433         DIC 1,DSKP      ;READ IT BACK
22 00615 122414         SUB# 1,0,SZR    ;CHECK REGISTER AND
23 00616 006253         EHALT           ;DATA PATHS THROUGH
24 00617 006254         LOOP            ;THE MUX'S
25
26 00620 006252 A12:    SETUP           ;SEE IF IORST WILL
27 00621 102200         ADC 0,0         ;CLEAR THE CA REGISTER
28 00622 062033         DOB 0,DSKP      ;LOAD IT WITH ALL 1'S
29 00623 062577         IORST           ;CLEAR IT TO ZEROS (RESET)
30 00624 065433         DIB 1,DSKP      ;READ IT BACK
31 00625 125004         MOV 1,1,SZR
32 00626 006253         EHALT
33 00627 006254         LOOP
34
35 00630 006252 A13:    SETUP           ;SEE IF IORST WILL
36 00631 102200         ADC 0,0         ;CLEAR DISK ADDRESS
37 00632 063033         DOC 0,DSKP      ;REGISTER, LOAD ALL 1'S
38 00633 062677         IORST           ;CLEAR TO ZEROS
39 00634 066433         DIC 1,DSKP      ;READ BACK
40 00635 125004         MOV 1,1,SZR     ;(S1,S2,S4,S8,ARE CLEARED
41 00636 006253         EHALT           ;VIA "RESET" THRU "RESET 3")
42 00637 006254         LOOP
43
```

```
10012 .MAIN
#1              ;THE FOLLOWING TEST REPLACES TESTS A14 TO A29
#2              ;TEST OF SINGLE BIT LOADS INTO CA (DOB 0,DSKP)
#3              ;AT EHALT: AC0 WILL=GOOD AC1 WILL=BAD
#4 00640 102520         SUBZL 0,0       ;START BIT 15
#5 00641 040137         STA 0,TEMP
#6 00642 006252 A14,29: SETUP
#7 00643 020137         LDA 0,TEMP
#8 00644 062033         DOB 0,DSKP
#9 00645 065433         DIB 1,DSKP      ;GET SINGLE BIT BACK
#10 00646 122414        SUB# 1,0,SZR    ;SKP=NOT ERROR
#11 00647 006253        EHALT
#12 00650 006254        LOOP            ;ITERATE SINGLE BIT
#13 00651 020137        LDA 0,TEMP
#14 00652 101124        MOVZL 0,0,SZR   ;SKP=DONE 16 BITS
#15 00653 000756        JMP A14,29-1    ;DO ONE MORE
#16
#17             ;THE FOLLOWING TEST REPLACES TESTS A30 TO A45
#18             ;TESY OF SINGLE 0 BIT LOADS INTO THE CA (DOB 0,DSKP)
#19 00654 102120         ADCZL 0,0      ;START #177776
#20 00655 040137         STA 0,TEMP
#21 00656 006252 A30,45: SETUP
#22 00657 020137         LDA 0,TEMP
#23 00660 062033         DOB 0,DSKP     ;OUT SINGLE NO BIT
#24 00661 065433         DIB 1,DSKP
#25 00662 122414         SUB# 1,0,SZR   ;OUT=IN? SKP
#26 00663 006253         EHALT
#27 00664 006254         LOOP           ;ITERATE SINGLE NO BIT
#28 00665 020137         LDA 0,TEMP
#29 00666 101142         MOVOL 0,0,SZC  ;CRY=0 IS DONE 16
#30 00667 000766         JMP A30,45-1   ;DO NEXT BIT
```

```
10013 .MAIN
01                      ;THE FOLLOWING TEST REPLACES TESTS A46 TO A61
02                      ;TEST OF SINGLE BIT LOADS INTO DSK ADRS (DOC 0,DSKP)
03                      ;AT EHALT: AC0 =ILL=GOOD AC1 =ILL=BAD
04 000670 102520                SUBZL 0,0               ;START BIT 15
05 000671 040137                STA 0,TEMP
06 000672 006252 A46,61:        SETUP
07 000673 020137                LDA 0,TEMP
08 000674 063033                DOC 0,DSKP
09 000675 066433                DIC 1,DSKP              ;GET SINGLE BIT BACK
10 000676 122414                SUB# 1,0,SZR            ;SKP=NOT ERROR
11 000677 006253                EHALT
12 000700 006254                LOOP                    ;ITERATE SINGLE BIT
13 000701 020137                LDA 0,TEMP
14 000702 101124                MOVZL 0,0,SZR           ;SKP=DONE 16 BITS
15 000703 000766                JMP A46,61-1            ;DO ONE MORE
16
17                      ;THE FOLLOWING TEST REPLACES TESTS A62 TO A77
18                      ;TEST OF SINGLE 0 BIT LOADS INTO THE DSK ADRS (DOC 0,DSKP)
19 000704 102120                ADCZL 0,0               ;START #177776
20 000705 040137                STA 0,TEMP
21 000706 006252 A62,77: SETUP
22 000707 020137                LDA 0,TEMP
23 000710 063033                DOC 0,DSKP              ;OUT SINGLE NO BIT
24 000711 066433                DIC 1,DSKP
25 000712 122414                SUB# 1,0,SZR            ;OUT=IN? SKP
26 000713 006253                EHALT
27 000714 006254                LOOP                    ;ITERATE SINGLE NO BIT
28 000715 020137                LDA 0,TEMP
29 000716 101142                MOVOL 0,0,SZC           ;CRY=0 IS DONE 16
30 000717 000766                JMP A62,77-1            ;DO NEXT BIT
31
32                      ;BIT#   BIT     NOBIT   BIT NAME
33                      ;15     000001  177776  SC1
34                      ;14     000002  177775  SC2
35                      ;13     000004  177773  SC4
36                      ;12     000010  177767  SC8
37                      ;11     000020  177757  S1
38                      ;10     000040  177737  S2
39                      ;9      000100  177677  S4
40                      ;8      000200  177577  S8
41                      ;7      000400  177377  HD1
42                      ;6      001000  176777  HD2
43                      ;5      002000  175777  HD4
44                      ;4      004000  173777  HD8
45                      ;3      010000  167777  HD16
46                      ;2      020000  157777  FORMAT
47                      ;1      040000  137777  D1
48                      ;0      100000  077777  D0
49
```

```
10014 .MAIN
01
02 000720 102400 A78:   SUB 0,0                 ;CHECK CA REGISTER, ALL
03 000721 006250                JSR @ISET               ;POSSIBLE PATTERNS
04 000722 062033                DOB 0,DSKP              ;LOAD CA
05 000723 065433                DIB 1,DSKP              ;READ IT BACK
06 000724 122414                SUB# 1,0,SZR            ;AC0=GOOD
07 000725 006253                EHALT                   ;AC1=BAD
08 000726 006254                LOOP                    ;DO IT ONLY ONCE FOR EACH PAT
09 000727 101404                INC 0,0,SZR             ;NEXT PATTERN
10 000730 000771                JMP .-7
11
12 000731 006250 A79:   JSR @ISET               ;CHECK DISK ADDRESS REGISTER
13 000732 063033                DOC 0,DSKP              ;ALL POSSIBLE PATTERNS
14 000733 066433                DIC 1,DSKP              ;LOAD/READ BACK
15 000734 122414                SUB# 1,0,SZR            ;AC0=GOOD
16 000735 006253                EHALT                   ;AC1=BAD
17 000736 006254                LOOP
18 000737 101404                INC 0,0,SZR             ;NEXT PATTERN
19 000740 000771                JMP .-7
20
21 000741 006252 A80:   SETUP                   ;SPECIAL TEST FOR 8291
22 000742 102402                SUB 0,0                 ;IC PACKS.  SET RIGHTMOST
23 000743 030127                LDA 2,BIT1              ;BIT IN EACH PACK WITH
24 000744 072033                DOB 2,DSKP              ;3 SUCCESSIVE LOADS, THEN
25 000745 072033                DOB 2,DSKP              ;LOAD ZERO.  MARGINAL
26 000746 072033                DOB 2,DSKP              ;PACKS WILL HOLD THE "1".
27 000747 062033                DOB 0,DSKP
28 000750 065433                DIB 1,DSKP              ;TESTING CA REGISTER
29 000751 125004                MOV 1,1,SZR             ;AC0=GOOD
30 000752 006253                EHALT                   ;AC1=BAD
31 000753 006254                LOOP
```

```
!0015 .MAIN
01
02 00754 006252 AB1:    SETUP           ;SPECIAL TEST FOR 8291
03 00755 102400          SUB 0,0         ;IC PACKS.  SET POSITION 2
04 00756 030130          LDA 2,BIT2      ;(PIN 9) IN EACH PACK WITH
05 00757 072033          DOB 2,DSKP      ;3 SUCCESSIVE LOADS, THEN
06 00760 072033          DOB 2,DSKP      ;LOAD ZEROS.  MARGINAL
07 00761 072033          DOB 2,DSKP      ;PACKS WILL HOLD A "1"
08 00762 062033          DOB 0,DSKP
09 00763 065433          DIB 1,DSKP      ;TESTING CA REGISTER
10 00764 122414          SUB# 1,0,SZR    ;AC0=GOOD
11 00765 006253          EHALT           ;AC1=BAD
12 00766 006254          LOOP
13
14 00767 006252 AB2:    SETUP           ;SPECIAL TEST FOR 8291
15 00770 102400          SUB 0,0         ;IC PACKS.  SET POSITION 4
16 00771 030131          LDA 2,BIT4      ;(PIN 2) IN EACH PACK WITH
17 00772 072033          DOB 2,DSKP      ;3 SUCCESSIVE LOADS, THEN
18 00773 072033          DOB 2,DSKP      ;LOAD ZEROS.  MARGINAL
19 00774 072033          DOB 2,DSKP      ;PACKS WILL HOLD A "1".
20 00775 062033          DOB 0,DSKP
21 00776 065433          DIB 1,DSKP      ;TESTING CA REGISTER
22 00777 122414          SUB# 1,0,SZR    ;AC0=GOOD
23 01000 006253          EHALT           ;AC1=BAD
24 01001 006254          LOOP
25
26 01002 006252 AB3:    SETUP           ;SPECIAL TEST FOR 8291
27 01003 102400          SUB 0,0         ;IC PACKS.  SET POSITION 8
28 01004 030132          LDA 2,BIT8      ;(PIN 12) IN EACH PACK WITH
29 01005 072033          DOB 2,DSKP      ;3 SUCCESSIVE LOADS.  THEN
30 01006 072033          DOB 2,DSKP      ;LOAD ZEROS.  MARGINAL
31 01007 072033          DOB 2,DSKP      ;PACKS WILL HOLD A "1"
32 01010 062033          DOB 0,DSKP
33 01011 065433          DIB 1,DSKP      ;TESTING CA REGISTER
34 01012 122414          SUB# 1,0,SZR    ;AC0=GOOD
35 01013 006253          EHALT           ;AC1=BAD
36 01014 006254          LOOP
37
38 01015 006252 AB4:    SETUP           ;SPECIAL TEST FOR 8291
39 01016 102400          SUB 0,0         ;IC PACKS.  SET POSITION 1
40 01017 030127          LDA 2,BIT1      ;(PIN 5) OF EACH PACK WITH
41 01020 073033          DOC 2,DSKP      ;3 SUCCESSIVE LOADS, THEN
42 01021 073033          DOC 2,DSKP      ;LOAD ZEROS.  MARGINAL
43 01022 073033          DOC 2,DSKP      ;PACKS WILL HOLD A "1".
44 01023 063033          DOC 0,DSKP
45 01024 066433          DIC 1,DSKP      ;TESTING DISK ADDR REG.
46 01025 122414          SUB# 1,0,SZR    ;AC0=GOOD
47 01026 006253          EHALT           ;AC1=BAD
48 01027 006254          LOOP
```

```
!0016 .MAIN
01
02 01030 006252 AB5:    SETUP           ;SPECIAL TEST FOR 8291
03 01031 102400          SUB 0,0         ;IC PACKS.  SET POSITION 2
04 01032 030130          LDA 2,BIT2      ;(PIN 9) OF EACH PACK WITH
05 01033 073033          DOC 2,DSKP      ;3 SUCCESSIVE LOADS, THEN
06 01034 073033          DOC 2,DSKP      ;LOAD ZEROS.  MARGINAL
07 01035 073033          DOC 2,DSKP      ;PACKS WILL RETAIN A "1".
08 01036 063033          DOC 0,DSKP
09 01037 066433          DIC 1,DSKP      ;TESTING DISK ADDR REG.
10 01040 122414          SUB# 1,0,SZR    ;AC0=GOOD
11 01041 006253          EHALT           ;AC1=BAD
12 01042 006254          LOOP
13
14 01043 006252 AB6:    SETUP           ;SPECIAL TEST FOR 8291
15 01044 102400          SUB 0,0         ;IC PACKS.  SET POSITION 4
16 01045 030131          LDA 2,BIT4      ;(PIN 2) OF EACH PACK WITH
17 01046 073033          DOC 2,DSKP      ;3 SUCCESSIVE LOADS, THEN
18 01047 073033          DOC 2,DSKP      ;LOAD ZEROS.  MARGINAL
19 01050 073033          DOC 2,DSKP      ;PACKS WILL RETAIN A "1".
20 01051 063033          DOC 0,DSKP
21 01052 066433          DIC 1,DSKP      ;TESTING DISK ADDR REG.
22 01053 122414          SUB# 1,0,SZR    ;AC0=GOOD
23 01054 006253          EHALT           ;AC1=BAD
24 01055 006254          LOOP
25
26 01056 006252 AB7:    SETUP           ;SPECIAL TEST FOR 8291
27 01057 102400          SUB 0,0         ;IC PACKS.  SET POSITION 8
28 01060 030132          LDA 2,BIT8      ;(PIN 12) OF EACH PACK WITH
29 01061 073033          DOC 2,DSKP      ;3 SUCCESSIVE LOADS, THEN
30 01062 073033          DOC 2,DSKP      ;LOAD ALL ZEROS.  MARGINAL
31 01063 073033          DOC 2,DSKP      ;PACKS WILL RETAIN A "1".
32 01064 063033          DOC 0,DSKP
33 01065 066433          DIC 1,DSKP      ;TESTING DISK ADDR REG.
34 01066 122414          SUB# 1,0,SZR    ;AC0=GOOD
35 01067 006253          EHALT           ;AC1=BAD
36 01070 006254          LOOP
37
38 01071 020142 B1:      LDA 0,KDOB      ;CHECK FOR ILLEGAL
39 01072 040403          STA 0,B1.1      ;DEVICE SELECT BY
40 01073 024133          LDA 1,C2525     ;DISK PACK CONTROL
41 01074 006251          SETP1           ;PERFORM "DOB" TO EVERY
42 01075 060000 B1.1:    DOB 1,0         ;OTHER DEVICE ADDRESS
43 01076 061433          DIB 0,DSKP      ;AND CHECK DSKP
44 01077 122415          SUB# 1,0,SNR    ;EACH TIME TO SEE
45 01100 006253          EHALT           ;IT RECOGNIZED THE "DOB".
46 01101 006254          LOOP
47 01102 010773 B1.2:    ISZ B1.1        ;CODE FROM HERE ON IS
48 01103 020772          LDA 0,B1.1      ;FOR INCREMENTING TO
49 01104 030116          LDA 2,C77       ;NEXT DEVICE ADDRESS
50 01105 143405          AND 2,0,SNR
51 01106 000405          JMP B2
52 01107 030147          LDA 2,CDSK
53 01110 142415          SUB# 2,0,SNR
54 01111 000771          JMP B1.2
55 01112 000762          JMP B1.1-1
```

```
10017 .MAIN
01
02 01113 006252 B2:     SETUP           ;SEE IF (S) PULSE WILL SET
03 01114 020063         LDA 0,C1000     ;"DP BUSY".  START A
04 01115 061133         DOAS 0,DSKP     ;SEEK
05 01116 063433         SKPBN DSKP      ;SKIP IF "DP BUSY" = 1
06 01117 006263         EHLT            ;CHECK "DP START", "DP BUSY",
07 01120 006273         LOOPD           ;"SELB" OC GATE.
08
09 01121 006252 B3:     SETUP           ;SEE IF (C) PULSE WILL CLEAR
10 01122 020063         LDA 0,C1000     ;"DP BUSY".  START A SEEK,
11 01123 061133         DOAS 0,DSKP     ; THEN CLEAR
12 01124 060233         NIOC DSKP
13 01125 063533         SKPBZ DSKP      ;CHECK "CLEAR"
14 01126 006263         EHLT
15 01127 006273         LOOPD
16
17 01130 006252 B4:     SETUP           ;SEE IF IORST WILL CLEAR
18 01131 020063         LDA 0,C1000     ;"DP BUSY".  START A SEEK,
19 01132 061133         DOAS 0,DSKP     ; THEN CLEAR IT
20 01133 062677         IORST
21 01134 063533         SKPBZ DSKP      ;CHECK "RESET", "CLEAR"
22 01135 006263         EHLT
23 01136 006273         LOOPD
24
25 01137 006252 B5:     SETUP           ;CHECK TO INSURE THAT
26 01140 064433         DIA 1,DSKP      ;ALL SEEKING FF'S ARE
27 01141 022223         LDA 0,C3600     ;CLEARED BY IORST
28 01142 107404         AND 0,1,SZR     ;AC1=BAD SEEKING.
29 01143 006253         EHALT           ;STATUS FROM DIA.
30 01144 006254         LOOP            ;CHECK DATA THRU MUX
31
32 01145 006252 B6:     SETUP           ;CHECK FOR PROPER
33 01146 102000         ADC 0,0         ;INPUT OF STATUS THRU
34 01147 062033         DOB 0,DSKP      ;THE MULTIPLEXORS TO
35 01150 063833         DOC 0,DSKP      ;THE DATA BUSS OC GATES
36 01151 064433         DIA 1,DSKP
37 01152 022223         LDA 0,C3600
38 01153 107414         AND# 0,1,SZR
39 01154 006253         EHALT           ;SEEKING STATUS ON
40 01155 006254         LOOP            ;ALL SHOULD BE ZERO, (AC1)=STATUS
41
42 01156 006252 B7:     SETUP           ;TRY TO SET "SEEKING 0"
43 01157 020063         LDA 0,C1000     ;VIA (S) PULSE
44 01160 061133         DOAS 0,DSKP
45 01161 064433         DIA 1,DSKP      ;"DP START" SETS "START"
46 01162 022064         LDA 0,KB5       ;CHECK "ADAPTER SEL", "SEEK"
47 01163 123415         AND# 1,0,SNR    ;AND "START SEEK"
48 01164 006263         EHLT
49 01165 006273         LOOPD
```

```
10018 .MAIN
01
02 01166 006252 B8:     SETUP           ;TRY TO SET "SEEKING 0"
03 01167 020063         LDA 0,C1000     ;WITH A (P) PULSE
04 01170 061333         DOAP 0,DSKP
05 01171 024064         LDA 1,KB5       ;CHECK "DP IOP"
06 01172 060433         DIA 0,DSKP      ;SET LINE TO "START"
07 01173 123415         AND# 1,0,SNR
08 01174 006263         EHLT
09 01175 006273         LOOPD
10
11 01176 006252 B9:     SETUP           ;CHECK UNIT SELECTION
12 01177 020063         LDA 0,C1000     ;START UNIT 0 SEEKING
13 01200 061333         DOAP 0,DSKP     ;AND VERIFY THAT NO
14 01201 072433         DIA 2,DSKP      ;OTHER UNIT SEEKS.
15 01202 020064         LDA 0,KB5       ;AC2=STATUS DURING SEEK
16 01203 024223         LDA 1,C3600     ;AC1=BAD SEEKING STATUS
17 01204 147400         AND 2,1         ;AC0=GOOD
18 01205 122414         SUB# 1,0,SZR    ;CHECK UNIT # DECODER
19 01206 006263         EHLT
20 01207 006273         LOOPD
21
22 01210 006252 B10:    SETUP           ;CHECK UNIT SELECTION 1
23 01211 020047         LDA 0,KB1       ;START UNIT 1 SEEKING
24 01212 063833         DOC 0,DSKP      ;AND VERIFY THAT NO
25 01213 020063         LDA 0,C1000     ;OTHER UNIT SEEKS.
26 01214 061333         DOAP 0,DSKP     ;AC2=STATUS DURING SEEK
27 01215 070433         DIA 2,DSKP      ;AC1=BAD SEEKING STATUS
28 01216 024223         LDA 1,C3600     ;AC0=GOOD
29 01217 020065         LDA 0,KB6       ;CHECK UNIT # DECODER,
30 01220 147400         AND 2,1         ;"UNIT 1", AND "SEEKING 1"
31 01221 122414         SUB# 1,0,SZR
32 01222 006263         EHLT
33 01223 006273         LOOPD
34
35 01224 006252 B11:    SETUP           ;CHECK UNIT SELECTION
36 01225 102020         SUBZR 0,0       ;START UNIT 2 SEEKING AND
37 01226 063833         DOC 0,DSKP      ;VERIFY THAT NO OTHER
38 01227 020063         LDA 0,C1000     ;UNIT SEEKS
39 01230 061333         DOAP 0,DSKP     ;AC2=STATUS DURING SEEK
40 01231 070433         DIA 2,DSKP      ;AC1=BAD SEEKING STATUS
41 01232 024223         LDA 1,C3600     ;AC0=GOOD
42 01233 020062         LDA 0,KB7       ;CHECK "UNIT 2",
43 01234 147400         AND 2,1         ;"SEEKING 2"
44 01235 122414         SUB# 1,0,SZR
45 01236 006263         EHLT
46 01237 006273         LOOPD
```

```
1#019 .MAIN
#1
#2 01244 #06252 B12:    SETUP           ;CHECK UNIT SELECTION
#3 01241 #20351         LDA #,C140K     ;START UNIT 3 SEEKING
#4 01242 #63333         DOC #,DSKP      ;AND VERIFY THAT NO
#5 01243 #22363         LDA #,C1200     ;OTHER UNIT SEEKS.
#6 01244 #61333         DOAP #,DSKP     ;AC2=STATUS DURING SEEK
#7 01245 #72433         DIA 2,DSKP      ;AC1=BAD SEEKING STATUS
#8 01246 #24223         LDA 1,C3600     ;AC0=GOOD
#9 01247 #20361         LDA #,K88       ;CHECK "UNIT 3", AND
10 01250 147400         AND 2,1         ;"SEEKING 3"
11 01251 122414         SUB# 1,0,SZR
12 01252 #06263         EHLT
13 01253 #06273         LOOPD
14
15 01254 #06252 B13:    SETUP           ;CHECK THE "CLEAR"
16 01255 152520         SUBZL 2,2       ;RESET OF "SEEKING 0".
17 01256 #06241         SSEEK           ;START UNIT 0 SEEKING
18 01257 #00233         NIOC DSKP       ;ISSUE (C) PULSE
19 01260 #22064         LDA #,K85
20 01261 #64433         DIA 1,DSKP      ;READ STATUS
21 01262 107404         AND #,1,SZR     ;"SEEKING 0" BIT NOT
22 01263 #06263         EHLT            ;CLEARED BY (C) PULSE
23 01264 #06273         LOOPD
24
25 01265 #06252 B14:    SETUP           ;CHECK THE "CLEAR"
26 01266 #30353         LDA 2,KB14      ;RESET OF "SEEKING 1"
27 01267 #06241         SSEEK           ;START UNIT 1 SEEKING.
28 01270 #00233         NIOC DSKP       ;ISSUE (C) PULSE
29 01271 #20263         LDA #,K86       ;READ STATUS
30 01272 #64433         DIA 1,DSKP      ;"SEEKING 1" BIT NOT
31 01273 107404         AND #,1,SZR     ;CLEARED BY (C) PULSE
32 01274 #06263         EHLT
33 01275 #06273         LOOPD
34
35 01276 #06252 B15:    SETUP           ;CHECK THE "CLEAR" RESET
36 01277 #30054         LDA 2,KB13      ;OF "SEEKING 2".
37 01300 #06241         SSEEK           ;START UNIT 2 SEEKING.
38 01301 #02233         NIOC DSKP       ;ISSUE (C) PULSE
39 01302 #20062         LDA #,KB7
40 01303 #64433         DIA 1,DSKP      ;READ STATUS
41 01304 107424         AND #,1,SZR     ;"SEEKING 2" BIT NOT
42 01305 #06263         EHLT            ;RESET BY (C) PULSE
43 01306 #06273         LOOPD
44
```

```
1#020 .MAIN
#1
#2 01307 #06252 B16:    SETUP           ;CHECK THE "CLEAR" RESET
#3 01310 #30355         LDA 2,KB12      ;OF "SEEKING 3"
#4 01311 #06241         SSEEK           ;START UNIT 3 SEEKING.
#5 01312 #60233         NIOC DSKP       ;ISSUE (C) PULSE
#6 01313 #20361         LDA #,K88
#7 01314 #64433         DIA 1,DSKP      ;READ STATUS
#8 01315 107404         AND #,1,SZR     ;"SEEKING 3" BIT NOT
#9 01316 #06263         EHLT            ;CLEARED BY (C) PULSE
10 01317 #06273         LOOPD
11
12 01320 #20144 B17:    LDA #,NOSKS     ;ATTEMPT TO OBTAIN
13 01321 101203         MOVR #,#,SNC    ;"SEEK DONE #" FROM
14 01322 #00415         JMP B18         ;A RECALIBRATE
15 01323 #06250         JSR #ISET       ;(SKIP OVER IF NO UNIT #)
16 01324 #06242         RECL#           ;RECAL UNIT #
17 01325 #20347         LDA #,KB1       ;AC1=STATUS
18 01326 123414         AND# 1,0,SZR    ;"ATTEN#" DID NOT SET
19 01327 #00403         JMP .+3
20 01330 #06263         EHLT
21 01331 #00405         JMP .+5
22 01332 #30225         LDA 2,C74K      ;CHECKN FOR SHORTED
23 01333 133400         AND 1,2         ;ATTENTION LINES. ON A HALT
24 01334 112404         SUB #,2,SZR     ;CK IF OTHER DONE FLOPS SET WITH
25 01335 #06263         EHLT            ;"SEEK DONE #"
26 01336 #06254         LOOP
27
28 01337 #20144 B18:    LDA #,NOSKS     ;ATTEMPT TO OBTAIN
29 01340 101200         MOVR #,#        ;"SEEK DONE 1" FROM
30 01341 101203         MOVR #,#,SNC    ;A RECALIBRATE
31 01342 #00415         JMP B19         ;(SKIP OVER IF NO UNIT 1)
32 01343 #06250         JSR #ISET
33 01344 #06243         RECL1           ;RECAL UNIT 1
34 01345 #20067         LDA #,KB2       ;AC1=STATUS
35 01346 123414         AND# 1,0,SZR    ;"ATTEN1" DID NOT SET
36 01347 #00403         JMP .+3
37 01350 #06263         EHLT
38 01351 #00405         JMP .+5
39 01352 #30225         LDA 2,C74K      ;CHECKN FOR SHORTED
40 01353 133400         AND 1,2         ;ATTENTION LINES. ON A HALT
41 01354 112404         SUB #,2,SZR     ;CK IF OTHER DONE FLOPS SET WITH
42 01355 #06263         EHLT            ;"SEEK DONE 1"
43 01356 #06254         LOOP
```

```
10021 .MAIN
01 01357 020144 B19:    LDA 0,NDSKS     ;ATTEMPT TO OBTAIN
02 01360 024054         LDA 1,KB13      ;"SEEK DONE 2" FROM
03 01361 123405         AND 1,0,SNR     ;A RECALIBRATE
04 01362 000415         JMP B20         ;(SKIP OVER IF NO UNIT 2)
05 01363 006250         JSR @ISET
06 01364 006244         RECL2           ;RECAL UNIT 2
07 01365 020056         LDA 0,KB3       ;AC1=STATUS
08 01366 123414         AND# 1,0,SZR    ;"ATTEN2" DID NOT SET
09 01367 000403         JMP .+3
10 01370 006263         EHLT
11 01371 000405         JMP .+5
12 01372 030225         LDA 2,C74K      ;CHECKN FOR SHORTED
13 01373 133400         AND 1,2         ;ATTENTION LINES. ON A HALT
14 01374 112404         SUB 0,2,SZR     ;CK IF OTHER DONE FLOPS SET WITH
15 01375 006263         EHLT            ;"SEEK DONE 2"
16 01376 006254         LOOP
17
18 01377 020144 B20:    LDA 0,NDSKS     ;ATTEMPT TO OBTAIN
19 01400 024055         LDA 1,KB12      ;"SEEK DONE 3" FROM
20 01401 123405         AND 1,0,SNR     ;A RECALIBRATE
21 01402 000415         JMP B21         ;(SKIP OVER IF NO UNIT 3)
22 01403 006250         JSR @ISET
23 01404 006245         RECL3           ;RECAL UNIT 3
24 01405 020055         LDA 0,KB4       ;AC1=STATUS
25 01406 123414         AND# 1,0,SZR    ;"ATTEN3" FAILED TO SET
26 01407 000403         JMP .+3
27 01410 006263         EHLT
28 01411 000405         JMP .+5
29 01412 030225         LDA 2,C74K      ;CHECKN FOR SHORTED
30 01413 133400         AND 1,2         ;ATTENTION LINES. ON A HALT
31 01414 112404         SUB 0,2,SZR     ;CK IF OTHER DONE FLOPS SET WITH
32 01415 006263         EHLT            ;"SEEK DONE 3"
33 01416 006254         LOOP
34
35 01417 062577 B21:    IORST           ;DISK DRIVE WILL NOT TAKE
36 01420 030162         LDA 2,C5        ;SUCCESSIVE RECALIBRATES
37 01421 006240         WAIT            ;DO A DELAY
38
39 01422 020144         LDA 0,NDSKS     ;SEE IF "SEEK DONE 0"
40 01423 101203         MOVR 0,0,SNC    ;WILL RESET "SEEKING 0"
41 01424 000407         JMP B22         ;(SKIP IF NO UNIT 0)
42 01425 006250         JSR @ISET
43 01426 006242         RECL0           ;RECAL UNIT 0
44 01427 020054         LDA 0,KB5       ;AC1=STATUS
45 01430 123414         AND# 1,0,SZR    ;"SEEKING 0" SHOULD GET
46 01431 006263         EHLT            ;RESET BY "SEEK DONE 0"
47 01432 006254         LOOP
48
49 01433 020144 B22:    LDA 0,NDSKS     ;SEE IF "SEEK DONE 1"
50 01434 101200         MOVR 0,0        ;WILL RESET "SEEKING 1"
51 01435 101203         MOVR 0,0,SNC
52 01436 000407         JMP B23         ;(SKIP OVER IF NO UNIT 1)
53 01437 006250         JSR @ISET
54 01440 006243         RECL1           ;RECAL UNIT 1
55 01441 000263         LDA 0,KB6       ;AC1=STATUS
56 01442 123414         AND# 1,0,SZR    ;"SEEKING 1" SHOULD GET
57 01443 006263         EHLT            ;RESET BY "SEEK DONE 1"
58 01444 006254         LOOP
```

```
10022 .MAIN
01
02 01445 020144 B23:    LDA 0,NDSKS     ;SEE IF "SEEK DONE 2"
03 01446 024054         LDA 1,KB13      ;WILL RESET "SEEKING 2"
04 01447 123405         AND 1,0,SNR
05 01450 000407         JMP B24         ;(SKIP OVER IF NO UNIT 2)
06 01451 006250         JSR @ISET
07 01452 006244         RECL2           ;RECAL UNIT 2
08 01453 020052         LDA 0,KB7       ;AC1=STATUS
09 01454 123414         AND# 1,0,SZR    ;"SEEKING 2" SHOULD GET
10 01455 006263         EHLT            ;RESET BY "SEEK DONE 2"
11 01456 006254         LOOP
12
13 01457 020144 B24:    LDA 0,NDSKS     ;SEE IF "SEEK DONE 3"
14 01460 024055         LDA 1,KB12      ;WILL RESET "SEEKING 3"
15 01461 123405         AND 1,0,SNR
16 01462 000407         JMP B25         ;(SKIP IF NO UNIT 3)
17 01463 006250         JSR @ISET
18 01464 006245         RECL3           ;RECAL UNIT 3
19 01465 020061         LDA 0,KB8       ;AC1=STATUS
20 01466 123414         AND# 1,0,SZR    ;"SEEKING 3" SHOULD GET
21 01467 006263         EHLT            ;RESET VIA "SEEK DONE 3"
22 01470 006254         LOOP
23
24 01471 062577 B25:    IORST           ;DISK DRIVE WILL NOT TAKE
25 01472 030165         LDA 2,C15       ;SUCCESSIVE RECALIBRATES
26 01473 006240         WAIT            ;WAIT 1.3 SEC
27
28 01474 020144         LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 0"
29 01475 101203         MOVR 0,0,SNC    ;WITH "DATOA"
30 01476 000411         JMP B26         ;SKIP OVER IF NO UNIT 0
31 01477 006250         JSR @ISET
32 01500 006242         RECL0           ;SET "SEEK DONE 0" VIA
33 01501 022047         LDA 0,KB1       ;RECALIBRATE
34 01502 061033         DOA 0,DSKP      ;ATTEMPT RESET
35 01503 064433         DIA 1,DSKP      ;READ STATUS
36 01504 123414         AND# 1,0,SZR
37 01505 006263         EHLT            ;"SEEK DONE 0" RESET FAILED
38 01506 006254         LOOP
39
40 01507 020144 B26:    LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 1"
41 01510 101200         MOVR 0,0        ;WITH "DATOA"
42 01511 101203         MOVR 0,0,SNC
43 01512 000411         JMP B27         ;(SKIP OVER IF NO UNIT 1)
44 01513 006250         JSR @ISET
45 01514 006243         RECL1           ;SET "SEEK DONE 1"
46 01515 020057         LDA 0,KB2       ;WITH A RECALIBRATE
47 01516 061033         DOA 0,DSKP      ;ATTEMPT RESET
48 01517 064433         DIA 1,DSKP      ;READ STATUS
49 01520 123414         AND# 1,0,SZR
50 01521 006263         EHLT            ;"SEEK DONE 1" RESET FAILED
51 01522 006254         LOOP
```

```
;0023 .MAIN
01
02 01523 020144 B27:    LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 2"
03 01524 024054         LDA 1,KB13      ;WITH "DATOA"
04 01525 123405         AND 1,0,SNR
05 01526 000411         JMP B28         ;(SKIP OVER IF NO UNIT 2)
06 01527 006250         JSR @ISET
07 01530 006244         RECL2           ;SET "SEEK DONE 2"
08 01531 020056         LDA 0,KB3       ;WITH A RECALIBRATE
09 01532 061033         DOA 0,DSKP      ;ATTEMPT RESET
10 01533 064433         DIA 1,DSKP      ;READ STATUS
11 01534 123414         AND# 1,0,SZR
12 01535 006263         EHLT            ;"SEEK DONE 2" RESET FAILED
13 01536 006254         LOOP
14
15 01537 020144 B28:    LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 3"
16 01540 024055         LDA 1,KB12      ;WITH "DATOA"
17 01541 123405         AND 1,0,SNR
18 01542 000411         JMP B29         ;(SKIP OVER IF NO UNIT 3)
19 01543 006250         JSR @ISET
20 01544 006245         RECL3           ;SET "SEEK DONE 3"
21 01545 020055         LDA 0,KB4       ;WITH A RECALIBRATE
22 01546 061033         DOA 0,DSKP      ;ATTEMPT RESET
23 01547 064433         DIA 1,DSKP      ;READ STATUS
24 01550 123414         AND# 1,0,SZR
25 01551 006263         EHLT            ;"SEEK DONE 3" RESET FAILED
26 01552 006254         LOOP
27
28 01553 020144 B29:    LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 0"
29 01554 101203         MOVR 0,0,SNC    ;WITH A (C) PULSE
30 01555 000411         JMP B30         ;(SKIP OVER IF NO UNIT 0)
31 01556 006250         JSR @ISET
32 01557 006242         RECL0           ;SET "SEEK DONE 0" VIA RECAL
33 01560 020047         LDA 0,KB1
34 01561 062233         NIOC DSKP       ;ATTEMPT CLEAR
35 01562 064433         DIA 1,DSKP      ;READ STATUS
36 01563 123414         AND# 1,0,SZR
37 01564 006263         EHLT            ;"SEEK DONE 0" RESET FAILED
38 01565 006254         LOOP
39
40 01566 020144 B30:    LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 1"
41 01567 101200         MOVR 0,0        ;WITH A (C) PULSE
42 01570 101203         MOVR 0,0,SNC
43 01571 000411         JMP B31         ;(SKIP IF NO UNIT 1)
44 01572 006250         JSR @ISET
45 01573 006243         RECL1           ;SET "SEEK DONE 1" VIA RECAL
46 01574 020057         LDA 0,KB2
47 01575 062233         NIOC DSKP       ;ATTEMPT RESET
48 01576 064433         DIA 1,DSKP      ;READ STATUS
49 01577 123414         AND# 1,0,SZR
50 01600 006263         EHLT            ;"SEEK DONE 1" RESET FAILED
51 01601 006254         LOOP
```

```
;0024 .MAIN
01
02 01602 020144 B31:    LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 2"
03 01603 024054         LDA 1,KB13      ;WITH A (C) PULSE
04 01604 123405         AND 1,0,SNR
05 01605 000411         JMP B32         ;(SKIP OVER IF NO UNIT 2)
06 01606 006250         JSR @ISET
07 01607 006244         RECL2           ;SET "SEEK DONE 2" VIA RECAL
08 01610 020056         LDA 0,KB3
09 01611 062233         NIOC DSKP       ;ATTEMPT RESET
10 01612 064433         DIA 1,DSKP      ;READ STATUS
11 01613 123414         AND# 1,0,SZR
12 01614 006263         EHLT            ;"SEEK DONE 2" RESET FAILED
13 01615 006254         LOOP
14
15 01616 020144 B32:    LDA 0,NDSKS     ;ATTEMPT TO RESET "SEEK DONE 3"
16 01617 024055         LDA 1,KB12      ;WITH A (C) PULSE
17 01620 123405         AND 1,0,SNR
18 01621 000411         JMP B33         ;(SKIP OVER IF NO UNIT 3)
19 01622 006250         JSR @ISET
20 01623 006245         RECL3           ;SET "SEEK DONE 3" VIA RECAL
21 01624 020055         LDA 0,KB4
22 01625 062233         NIOC DSKP       ;ATTEMPT RESET
23 01626 064433         DIA 1,DSKP      ;READ STATUS
24 01627 123414         AND# 1,0,SZR
25 01630 006263         EHLT            ;"SEEK DONE 3" RESET FAILED
26 01631 006254         LOOP
27
25 01632 006252 B33:    SETUP           ;IDLE STATE STATUS CHECK
29 01633 064433         DIA 1,DSKP
30 01634 102020         SUBZR 0,0
31 01635 107414         AND# 0,1,SZR
32 01636 006253         EHALT           ;R/W DONE = 1
33 01637 006254         LOOP
34
35 01640 006252 B36:    SETUP           ;IDLE STATE STATUS CHECK
36 01641 064433         DIA 1,DSKP
37 01642 020170         LDA 0,C30
38 01643 123414         AND# 1,0,SZR
39 01644 006253         EHALT           ;ADDR ERR OR END CYL
40 01645 006254         LOOP
```

```
10025 .MAIN
01
02 01646 006252 B371:   SETUP           ;IDLE STATE STATUS CHECK
03 01647 064433         DIA 1,DSKP
04 01650 020054         LDA 0,KB13
05 01651 107414         AND# 0,1,SZR
06 01652 006253         EHALT           ;CHECK WORD ERROR = 1
07 01653 006254         LOOP
08
09 01654 006252 B381:   SETUP           ;IDLE STATE STATUS CHECK
10 01655 064433         DIA 1,DSKP
11 01656 022112         LDA 0,C3
12 01657 123414         AND# 1,0,SZR
13 01660 006253         EHALT           ;DATA LATE OR ERR
14 01661 006254         LOOP
15
16 01662 020144 B391:   LDA 0,NDSKS     ;SEE IF "SEEK DONE 0"
17 01663 101203         MOVR 0,0,SNC    ;WILL CAUSE INTERRUPT
18 01664 000405         JMP B40         ;(SKIP IF NO UNIT 0)
19 01665 006250         JSR @ISET
20 01666 006242         RECL0           ;SET "SEEK DONE 0"
21 01667 006246         ITRWT           ;IS THERE AN INTERRUPT?
22 01670 006263         EHLT            ;NO, CHECK "INTERRUPT",
23 01671 006254         LOOP            ;"DP INT REQ".
24
25 01672 020144 B401:   LDA 0,NDSKS     ;SEE IF "SEEK DONE 1"
26 01673 101200         MOVR 0,0        ;WILL CAUSE INTERRUPT
27 01674 101203         MOVR 0,0,SNC
28 01675 000405         JMP B41         ;(SKIP IF NO UNIT 1)
29 01676 006250         JSR @ISET
30 01677 006243         RECL1           ;SET "SEEK DONE 1"
31 01700 006246         ITRWT           ;IS THERE AN INTERRUPT ?
32 01701 006263         EHLT            ;NO, CHECK "INTERRUPT",
33 01702 006254         LOOP            ;"DP INT REQ".
34
35 01703 020144 B411:   LDA 0,NDSKS     ;SEE IF "SEEK DONE 2"
36 01704 024054         LDA 1,KB13      ;WILL CAUSE INTERRUPT
37 01705 123405         AND 1,0,SNR
38 01706 000406         JMP B42         ;(SKIP IF NO UNIT 2)
39 01707 006250         JSR @ISET
40 01710 006244         RECL2           ;SET "SEEK DONE 2"
41 01711 006246         ITRWT           ;IS THERE AN INTERRUPT ?
42 01712 006263         EHLT            ;NO, CHECK "INTERRUPT",
43 01713 006254         LOOP            ;"DP INT REQ".
44
45
```

```
10026 .MAIN
01
02 01714 020144 B42:    LDA 0,NDSKS     ;SEE IF "SEEK DONE 3"
03 01715 024055         LDA 1,KB12      ;WILL CAUSE INTERRUPT
04 01716 123405         AND 1,0,SNR
05 01717 000406         JMP B43         ;(SKIP IF NO UNIT 3)
06 01720 006250         JSR @ISET
07 01721 006245         RECL3           ;SET "SEEK DONE 3"
08 01722 006246         ITRWT           ;IS THERE AN INTERRUPT ?
09 01723 006263         EHLT            ;NO, CHECK "INTERRUPT",
10 01724 006254         LOOP            ;"DP INT REQ".
11
12 01725 006251 B43:    SETP1           ;TEST SEEK TO CYLINDER 0
13 01726 020145         LDA 0,TESTU     ;(FIRST ATTEMPT AT COMPLETE SEEK)
14 01727 063333         DOC 0,DSKP      ;SELECT AN ACTIVE UNIT #
15 01730 020063         LDA 0,KB6
16 01731 061333         DOAP 0,DSKP     ;SEEK 1
17 01732 032162         LDA 2,C5
18 01733 006240         WAIT            ;WAIT 500MS (OR UNTIL "DONE")
19 01734 020225         LDA 0,C74K      ;AC1=STATUS
20 01735 123415         AND# 1,0,SNR    ;NO SEEK DONE STATUS
21 01736 006253         EHALT           ;CHECK "SEEK","CONTROL 1","CYL",
22 01737 006254         LOOP            ;"HD+DIR","CONTROL 2", SEQUENCING.
23
24 01740 006251 B44:    SETP1           ;TEST SEEK TO CYLINDER 0
25 01741 020145         LDA 0,TESTU
26 01742 063333         DOC 0,DSKP      ;SELECT AN ACTIVE UNIT #
27 01743 020063         LDA 0,KB6
28 01744 061333         DOAP 0,DSKP     ;SEEK 1
29 01745 032162         LDA 2,C5
30 01746 006240         WAIT            ;WAIT 320MS (OR UNTIL "DONE")
31 01747 020057         LDA 0,KB10      ;AC1=STATUS
32 01750 123414         AND# 1,0,SZR    ;SEEK ERROR STATUS
33 01751 006253         EHALT           ;CHECK "SEEK","CONTROL 1","CYL",
34 01752 006254         LOOP            ;"HD+DIR", "DIFF","CONTROL 2".
35
36 01753 006251 B45:    SETP1           ;TEST SEEK CYLINDER 77 (OCTAL)
37 01754 006247         GADSK
38 01755 006257         SEEK
39 01756 000077         77
40 01757 020060         LDA 0,KB9       ;AC1=ENDING STATUS
41 01760 030057         LDA 2,KB10
42 01761 107414         AND# 0,1,SZR
43 01762 147414         AND# 2,1,SZR    ;SEEK ERROR OR
44 01763 006253         EHALT           ;NO READY STATUS
45 01764 006254         LOOP
46
47 01765 006251 B46:    SETP1           ;TEST SEEK CYLINDER 300 (OCTAL)
48 01766 006247         GADSK
49 01767 006257         SEEK
50 01770 000300         300
51 01771 020060         LDA 0,KB9       ;AC1=ENDING STATUS
52 01772 030057         LDA 2,KB10
53 01773 107414         AND# 0,1,SZR
54 01774 147414         AND# 2,1,SZR    ;SEEK ERROR OR
55 01775 006253         EHALT           ;NO READY STATUS
56 01776 006254         LOOP
```

```
!0027 .MAIN
#1
#2 01777 006251 B471:    SETP1          ;CHOOSE AN ACTIVE DISK
#3 02000 006247          GADSK          ;AND SET INTERRUPT VIA SEEK.
#4 02001 006257          SEEK           ;CHECK FOR PROPER DISK
#5 02002 000000          0              ;ADDRESS RESPONSE TO INTA
#6 02003 006246          ITRWT          ;ALLOW INTERRUPT
#7 02004 000401          JMP .+1
#8 02005 02#147          LDA 0,CDSK
#9 02006 065177          INTA 1         ;GET THE ADDRESS
10 02007 122414          SUB# 1,0,SZR   ;INTA FAILED.
11 02010 006253          EHALT          ;AC0=GOOD
12 02011 006254          LOOP           ;AC1=BAD
13
14 02012 006251 B481:    SETP1          ;SEE IF (C) PULSE WILL
15 02013 006247          GADSK          ;CLEAR "DP INT REQ"
16 02014 006257          SEEK           ;SET "INTERRUPT" WITH
17 02015 000000          0              ;A SEEK
18 02016 063233          NIOC DSKP      ;CLEAR INT REQ
19 02017 006246          ITRWT          ;CHECK FOR INTERRUPT
20 02020 000402          JMP .+2        ;NO INTERRUPT, OK!!
21 02021 006253          EHALT          ;(C) THRU "CLEAR ALL" FAILS TO
22 02022 006254          LOOP           ;CLEAR "DP INT REQ".
23
24 02023 006251 B491:    SETP1          ;INSURE THAT "DP INT DISABLE"
25 02024 102000          ADC 0,0
26 02025 040277          STA 0,SW
27 02026 063077          MSKO 0         ;SET DISABLE
28 02027 006247          GADSK          ;GET ACTIVE UNIT # IN (AC2)
29 02030 006257          SEEK           ;START A SEEK
30 02031 000000          0
31 02032 006246          ITRWT          ;IS THERE AN INTERRUPT ?
32 02033 000402          JMP .+2        ;NO, GOOD
33 02034 006253          EHALT          ;MSKO BIT 7 DID NOT SET "DP DISABLE"
34 02035 006254          LOOP           ;AND INHIBIT "DP INT REQ".
35
36 02036 006251 B501:    SETP1          ;INSURE THAT IORST WILL
37 02037 102000          ADC 0,0
38 02040 040277          STA 0,SW
39 02041 063077          MSKO 0         ;SET "DP INT DISABLE"
40 02042 063677          IORST          ;CLEAR IT
41 02043 006247          GADSK          ;GET ACTIVE UNIT # IN (AC2)
42 02044 006257          SEEK           ;START A SEEK
43 02045 000000          0
44 02046 006246          ITRWT          ;IS THERE AN INTERRUPT
45 02047 006253          EHALT          ;NO, CHECK RESET OF "DP INT
46 02050 006254          LOOP           ;DISABLE".
```

```
!0028 .MAIN
#1
#2 02051 020144 B511:    LDA 0,NDSKS    ;IF UNIT 0 NOT AVAILABLE
#3 02052 101203          MOVR 0,0,SNC   ;GO ON TO UNIT 1
#4 02053 000423          JMP B53        ;(UNIT 0 NOT BEING TESTED)
#5 02054 006252          SETUP          ;IDLE STATE STATUS CHECK
#6 02055 152520          SUBZL 2,2      ;SELECT THE ADAPTER
#7 02056 006257          SEEK           ;WITH A SEEK 0
#8 02057 000000          0
#9 02060 064433          DIA 1,DSKP     ;READ STATUS
10 02061 020060          LDA 0,KB9
11 02062 123415          AND# 1,0,SNR
12 02063 006253          EHALT          ;NO READY STATUS, UNIT 0
13 02064 006254          LOOP
14
15 02065 006252 B521:    SETUP          ;IDLE STATE STATUS CHECK
16 02066 152520          SUBZL 2,2      ;SELECT THE ADAPTER
17 02067 006257          SEEK           ;WITH A SEEK 0
18 02070 000000          0
19 02071 064433          DIA 1,DSKP     ;READ STATUS
20 02072 020057          LDA 0,KB10
21 02073 123414          AND# 1,0,SZR
22 02074 006253          EHALT          ;SEEK ERROR STATUS, UNIT 0
23 02075 006254          LOOP
24
25
26 02076 020144 B531:    LDA 0,NDSKS    ;IF UNIT 1 IS NOT
27 02077 101200          MOVR 0,0       ;AVAILABLE, GO ON TO UNIT 2
28 02100 101203          MOVR 0,0,SNC
29 02101 000423          JMP B55        ;SKIP THIS TEST
30 02102 006252          SETUP          ;IDLE STATE STATUS CHECK
31 02103 030053          LDA 2,KB14     ;SELECT THE ADAPTER
32 02104 006257          SEEK           ;WITH A SEEK 0
33 02105 000000          0
34 02106 064433          DIA 1,DSKP     ;READ STATUS
35 02107 020060          LDA 0,KB9
36 02110 123415          AND# 1,0,SNR
37 02111 006253          EHALT          ;NO READY STATUS, UNIT 1
38 02112 006254          LOOP
39
40 02113 006252 B541:    SETUP          ;IDLE STATE STATUS CHECK
41 02114 030053          LDA 2,KB14
42 02115 006257          SEEK           ;SELECT THE ADAPTER
43 02116 000000          0              ;WITH A SEEK 0
44 02117 064433          DIA 1,DSKP     ;READ STATUS
45 02120 020057          LDA 0,KB10
46 02121 123414          AND# 1,0,SZR
47 02122 006253          EHALT          ;SEEK ERROR STATUS, UNIT 1
48 02123 006254          LOOP
```

```
10029 .MAIN
01
02 02124 020144 B551:   LDA 0,NOSKS      ;IF UNIT 2 IS NOT AVAILABLE
03 02125 024054         LDA 1,KB13       ;GO ON TO UNIT 3
04 02126 107405         AND 0,1,SNR
05 02127 000423         JMP B57          ;SKIP THIS TEST, NO UNIT 2
06 02130 006252         SETUP            ;IDLE STATE STATUS CHECK
07 02131 030054         LDA 2,KB13       ;SELECT THE ADAPTER
08 02132 006257         SEEK             ;WITH A SEEK 0
09 02133 000000         0
10 02134 064433         DIA 1,DSKP       ;READ STATUS
11 02135 022050         LDA 0,KB9
12 02136 123415         ANDO# 1,0,SNR
13 02137 006253         EHALT            ;NO READY STATUS, UNIT 2
14 02140 006254         LOOP
15
16 02141 006252 B58:    SETUP            ;IDLE STATE STATUS CHECK
17 02142 030054         LDA 2,KB13       ;SELECT THE ADAPTER
18 02143 006257         SEEK             ;WITH A SEEK 0
19 02144 000000         0
20 02145 064433         DIA 1,DSKP       ;READ STATUS
21 02146 020057         LDA 0,KB10
22 02147 123414         ANDO# 1,0,SZR
23 02150 006253         EHALT            ;SEEK ERROR STATUS, UNIT 2
24 02151 006254         LOOP
25
26 02152 020144 B57:    LDA 0,NOSKS      ;IF UNIT 3 IS NOT AVAILABLE
27 02153 024055         LDA 1,KB12       ;GO TO NEXT TEST.
28 02154 123405         AND 1,0,SNR      ;
29 02155 000423         JMP D1           ;SKIP, NO UNIT 3
30 02156 006252         SETUP            ;IDLE STATE STATUS CHECK
31 02157 030055         LDA 2,KB12       ;SELECT THE ADAPTER
32 02160 006257         SEEK             ;WITH A SEEK 0
33 02161 000000         0
34 02162 064433         DIA 1,DSKP       ;READ STATUS
35 02163 020050         LDA 0,KB9
36 02164 123415         ANDO# 1,0,SNR
37 02165 006253         EHALT            ;NO READY STATUS, UNIT 3
38 02166 006254         LOOP
39
40 02167 006252 B58:    SETUP            ;IDLE STATE STATUS CHECK
41 02170 030055         LDA 2,KB12       ;SELECT THE ADAPTER
42 02171 006257         SEEK             ;WITH A SEEK 0
43 02172 000000         0
44 02173 064433         DIA 1,DSKP       ;READ STATUS
45 02174 020057         LDA 0,KB10
46 02175 123414         ANDO# 1,0,SZR
47 02176 006253         EHALT            ;SEEK ERROR STATUS, UNIT 3
48 02177 006254         LOOP
```

```
10030 .MAIN
01
02 02200 006252 D1:     SETUP            ;CHECK THE FIRST 2 DATA
03 02201 020145         LDA 0,TESTU      ;CHANNEL CYCLES OF A WRITE
04 02202 063033         DOC 0,DSKP       ;(FIRST ATTEMPT AT WRITE)
05 02203 030153         LDA 2,BUFF
06 02204 072033         DOB 2,DSKP       ;LOAD CA REG.
07 02205 020062         LDA 0,KB7
08 02206 061133         DOAS 0,DSKP      ;WRITE !!
09 02207 006160         JSR @STALL       ;ALLOW TIME FOR DCH CYCLES
10 02210 060233         NIOC DSKP        ;CLEAR THE WRITE OPERATION
11 02211 061433         DIB 0,DSKP       ;AC0=ENDING MEMORY ADDRESS
12 02212 024053         LDA 1,KB14       ;AC2=STARTING MEMORY ADDRESS
13 02213 147000         ADD 2,1          ;2 DCH CYCLES SHOULD HAVE
14 02214 122414         SUB# 1,0,SZR     ;OCCURRED
15 02215 006253         EHALT            ;AC0=BAD, AC1=GOOD
16 02216 006254         LOOP             ;CHECK THE SEQUENCE "DP FLAG"-
                                         ;"REQ1"-"2ND REQ",  "DP FLAG"
                                         ;SETS "DP DCH REQ" WHICH STAYS
                                         ;ON FOR 2 MEMORY CYCLES.
17
18
19
20
21 02217 006252 D2:     SETUP            ;CHECK THE FIRST 2 DATA
22 02220 020145         LDA 0,TESTU      ;CHANNEL CYCLES OF A WRITE
23 02221 063033         DOC 0,DSKP
24 02222 030153         LDA 2,BUFF
25 02223 072033         DOB 2,DSKP       ;SET MEM ADDR REG.
26 02224 024153         LDA 1,C2525
27 02225 045000         STA 1,0,2        ;DATA = 052525
28 02226 045001         STA 1,1,2        ; 2 WORDS
29 02227 020062         LDA 0,KB7
30 02230 061133         DOAS 0,DSKP      ;WRITE !!
31 02231 006160         JSR @STALL       ;ALLOW TIME FOR 2 DCH'S
32 02232 060233         NIOC DSKP        ;CLEAR THE WRITE
33 02233 021000         LDA 0,0,2        ;IS THE DATA STILL IN MEMORY
34 02234 031001         LDA 2,1,2        ;IF NOT A DCH MAY HAVE
35 02235 1R0415         SUB# 0,1,SNR     ;BEEN PERFORMED INSTEAD
36 02236 140414         SUB# 2,1,SZR     ;OF A DCHO
37 02237 006253         EHALT            ;AC0&AC2=BAD WORDS
38 02240 006254         LOOP             ;AC1=GOOD
```

```
IP031 .MAIN
#1
#2 02241 006252 D3:     SETUP           ;CHECK PROPER CA REGISTER
#3 02242 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
#4 02243 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
#5 02244 102400         SUB 0,0         ;CYCLES AT THE BEGINNING
#6 02245 062033         DOB 0,DSKP      ;OF A WRITE.
#7 02246 024062         LDA 1,KB7
#8 02247 065133         DOAS 1,DSKP     ; WRITE !!
#9 02250 006160         JSR @STALL
10 02251 060233         NIOC DSKP       ;STOP THE WRITE
11 02252 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
12 02253 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
13 02254 113000         ADD 0,2         ;     ADDRESS
14 02255 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
15 02256 006253         EHALT           ;     ADDRESS
16 02257 006254         LOOP
17
18 02260 006252 D4:     SETUP           ;CHECK PROPER CA REGISTER
19 02261 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
20 02262 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
21 02263 102520         SUBZL 0,0       ;CYCLES AT THE BEGINNING
22 02264 062033         DOB 0,DSKP      ;OF A WRITE.
23 02265 024062         LDA 1,KB7
24 02266 065133         DOAS 1,DSKP     ; WRITE !!
25 02267 006160         JSR @STALL
26 02270 060233         NIOC DSKP       ;STOP THE WRITE
27 02271 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
28 02272 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
29 02273 113000         ADD 0,2         ;     ADDRESS
30 02274 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
31 02275 006253         EHALT           ;     ADDRESS
32 02276 006254         LOOP
33
34 02277 006252 D5:     SETUP           ;CHECK PROPER CA REGISTER
35 02300 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
36 02301 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
37 02302 020112         LDA 0,C3        ;CYCLES AT THE BEGINNING
38 02303 062033         DOB 0,DSKP      ;OF A WRITE.
39 02304 024062         LDA 1,KB7
40 02305 065133         DOAS 1,DSKP     ; WRITE !!
41 02306 006160         JSR @STALL
42 02307 060233         NIOC DSKP       ;STOP THE WRITE
43 02310 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
44 02311 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
45 02312 113000         ADD 0,2         ;     ADDRESS
46 02313 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
47 02314 006253         EHALT           ;     ADDRESS
48 02315 006254         LOOP
```

```
IP032 .MAIN
#1
#2 02316 006252 D6:     SETUP           ;CHECK PROPER CA REGISTER
#3 02317 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
#4 02320 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
#5 02321 020113         LDA 0,C7        ;CYCLES AT THE BEGINNING
#6 02322 062033         DOB 0,DSKP      ;OF A WRITE.
#7 02323 024062         LDA 1,KB7
#8 02324 065133         DOAS 1,DSKP     ; WRITE !!
#9 02325 006160         JSR @STALL
10 02326 060233         NIOC DSKP       ;STOP THE WRITE
11 02327 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
12 02330 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
13 02331 113000         ADD 0,2         ;     ADDRESS
14 02332 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
15 02333 006253         EHALT           ;     ADDRESS
16 02334 006254         LOOP
17
18 02335 006252 D7:     SETUP           ;CHECK PROPER CA REGISTER
19 02336 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
20 02337 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
21 02340 020114         LDA 0,C17       ;CYCLES AT THE BEGINNING
22 02341 062033         DOB 0,DSKP      ;OF A WRITE.
23 02342 024062         LDA 1,KB7
24 02343 065133         DOAS 1,DSKP     ; WRITE !!
25 02344 006160         JSR @STALL
26 02345 060233         NIOC DSKP       ;STOP THE WRITE
27 02346 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
28 02347 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
29 02350 113000         ADD 0,2         ;     ADDRESS
30 02351 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
31 02352 006253         EHALT           ;     ADDRESS
32 02353 006254         LOOP
33
34 02354 006252 D8:     SETUP           ;CHECK PROPER CA REGISTER
35 02355 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
36 02356 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
37 02357 020115         LDA 0,C37       ;CYCLES AT THE BEGINNING
38 02360 062033         DOB 0,DSKP      ;OF A WRITE.
39 02361 024062         LDA 1,KB7
40 02362 065133         DOAS 1,DSKP     ; WRITE !!
41 02363 006160         JSR @STALL
42 02364 060233         NIOC DSKP       ;STOP THE WRITE
43 02365 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
44 02366 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
45 02367 113000         ADD 0,2         ;     ADDRESS
46 02370 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
47 02371 006253         EHALT           ;     ADDRESS
48 02372 006254         LOOP
```

```
10033 .MAIN
#1
#2 02373 006252 D9:     SETUP           ;CHECK PROPER CA REGISTER
#3 02374 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
#4 02375 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
#5 02376 020116         LDA 0,C77       ;CYCLES AT THE BEGINNING
#6 02377 062033         DOB 0,DSKP      ;OF A WRITE.
#7 02400 024062         LDA 1,KB7
#8 02401 065133         DOAS 1,DSKP     ; WRITE !!
#9 02402 006160         JSR @STALL
10 02403 062233         NIOC DSKP       ;STOP THE WRITE
11 02404 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
12 02405 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
13 02406 113000         ADD 0,2         ;     ADDRESS
14 02407 146414         SUBM 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
15 02410 006253         EHALT           ;     ADDRESS
16 02411 006254         LOOP
17
18 02412 006252 D10:    SETUP           ;CHECK PROPER CA REGISTER
19 02413 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
20 02414 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
21 02415 020117         LDA 0,C177      ;CYCLES AT THE BEGINNING
22 02416 062033         DOB 0,DSKP      ;OF A WRITE.
23 02417 024062         LDA 1,KB7
24 02420 065133         DOAS 1,DSKP     ; WRITE !!
25 02421 006160         JSR @STALL
26 02422 062233         NIOC DSKP       ;STOP THE WRITE
27 02423 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
28 02424 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
29 02425 113000         ADD 0,2         ;     ADDRESS
30 02426 146414         SUBM 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
31 02427 006253         EHALT           ;     ADDRESS
32 02430 006254         LOOP
33
34 02431 006252 D11:    SETUP           ;CHECK PROPER CA REGISTER
35 02432 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
36 02433 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
37 02434 020120         LDA 0,C377      ;CYCLES AT THE BEGINNING
38 02435 062033         DOB 0,DSKP      ;OF A WRITE.
39 02436 024062         LDA 1,KB7
40 02437 065133         DOAS 1,DSKP     ; WRITE !!
41 02440 006160         JSR @STALL
42 02441 062233         NIOC DSKP       ;STOP THE WRITE
43 02442 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
44 02443 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
45 02444 113000         ADD 0,2         ;     ADDRESS
46 02445 146414         SUBM 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
47 02446 006253         EHALT           ;     ADDRESS
48 02447 006254         LOOP
```

```
10034 .MAIN
#1
#2 02450 006252 D12:    SETUP           ;CHECK PROPER CA REGISTER
#3 02451 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
#4 02452 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
#5 02453 020121         LDA 0,C777      ;CYCLES AT THE BEGINNING
#6 02454 062033         DOB 0,DSKP      ;OF A WRITE.
#7 02455 024062         LDA 1,KB7
#8 02456 065133         DOAS 1,DSKP     ; WRITE !!
#9 02457 006160         JSR @STALL
10 02460 062233         NIOC DSKP       ;STOP THE WRITE
11 02461 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
12 02462 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
13 02463 113000         ADD 0,2         ;     ADDRESS
14 02464 146414         SUBM 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
15 02465 006253         EHALT           ;     ADDRESS
16 02466 006254         LOOP
17
18 02467 006252 D13:    SETUP           ;CHECK PROPER CA REGISTER
19 02470 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
20 02471 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
21 02472 020122         LDA 0,C1777     ;CYCLES AT THE BEGINNING
22 02473 062033         DOB 0,DSKP      ;OF A WRITE.
23 02474 024062         LDA 1,KB7
24 02475 065133         DOAS 1,DSKP     ; WRITE !!
25 02476 006160         JSR @STALL
26 02477 062233         NIOC DSKP       ;STOP THE WRITE
27 02500 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
28 02501 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
29 02502 113000         ADD 0,2         ;     ADDRESS
30 02503 146414         SUBM 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
31 02504 006253         EHALT           ;     ADDRESS
32 02505 006254         LOOP
33
34 02506 006252 D14:    SETUP           ;CHECK PROPER CA REGISTER
35 02507 020145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
36 02510 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
37 02511 020123         LDA 0,C3777     ;CYCLES AT THE BEGINNING
38 02512 062033         DOB 0,DSKP      ;OF A WRITE.
39 02513 024062         LDA 1,KB7
40 02514 065133         DOAS 1,DSKP     ; WRITE !!
41 02515 006160         JSR @STALL
42 02516 062233         NIOC DSKP       ;STOP THE WRITE
43 02517 030053         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
44 02520 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
45 02521 113000         ADD 0,2         ;     ADDRESS
46 02522 146414         SUBM 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
47 02523 006253         EHALT           ;     ADDRESS
48 02524 006254         LOOP
```

```
10035 .MAIN
01
02 02525 006252 D15:    SETUP           ;CHECK PROPER CA REGISTER
03 02526 022145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
04 02527 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
05 02530 022124         LDA 0,C7777     ;CYCLES AT THE BEGINNING
06 02531 062033         DOB 0,DSKP      ;OF A WRITE.
07 02532 024062         LDA 1,KB7
08 02533 065133         DOAS 1,DSKP     ; WRITE !!
09 02534 006160         JSR @STALL
10 02535 062233         NIOC DSKP       ;STOP THE WRITE
11 02536 030055         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
12 02537 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
13 02540 113000         ADD 0,2         ;     ADDRESS
14 02541 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
15 02542 006253         EHALT           ;     ADDRESS
16 02543 006254         LOOP
17
18 02544 006252 D16:    SETUP           ;CHECK PROPER CA REGISTER
19 02545 022145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
20 02546 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
21 02547 022125         LDA 0,C017      ;CYCLES AT THE BEGINNING
22 02550 062033         DOB 0,DSKP      ;OF A WRITE.
23 02551 024062         LDA 1,KB7
24 02552 065133         DOAS 1,DSKP     ; WRITE !!
25 02553 006160         JSR @STALL
26 02554 062233         NIOC DSKP       ;STOP THE WRITE
27 02555 030055         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
28 02556 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
29 02557 113000         ADD 0,2         ;     ADDRESS
30 02560 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
31 02561 006253         EHALT           ;     ADDRESS
32 02562 006254         LOOP
33
34 02563 006252 D17:    SETUP           ;CHECK PROPER CA REGISTER
35 02564 022145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
36 02565 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
37 02566 022126         LDA 0,C037      ;CYCLES AT THE BEGINNING
38 02567 062033         DOB 0,DSKP      ;OF A WRITE.
39 02570 024062         LDA 1,KB7
40 02571 065133         DOAS 1,DSKP     ; WRITE !!
41 02572 006160         JSR @STALL
42 02573 062233         NIOC DSKP       ;STOP THE WRITE
43 02574 030055         LDA 2,KB14      ;AC0=STARTING MEMORY ADDRESS
44 02575 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
45 02576 113000         ADD 0,2         ;     ADDRESS
46 02577 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
47 02600 006253         EHALT           ;     ADDRESS
48 02601 006254         LOOP
```

```
10036 .MAIN
01
02 02602 006252 D18:    SETUP           ;CHECK PROPER CA REGISTER
03 02603 022145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
04 02604 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
05 02605 102220         ADCZR 0,0       ;CYCLES AT THE BEGINNING
06 02606 062033         DOB 0,DSKP      ;OF A WRITE.
07 02607 024062         LDA 1,KB7
08 02610 065133         DOAS 1,DSKP     ; WRITE !!
09 02611 006160         JSR @STALL
10 02612 062233         NIOC DSKP       ;STOP THE WRITE
11                                      ;AC0=STARTING MEMORY ADDRESS
12 02613 065433         DIB 1,DSKP      ;AC1=ACTUAL ENDING MEMORY
13 02614 152520         SUBZL 2,2       ;     ADDRESS
14 02615 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
15 02616 006253         EHALT           ;     ADDRESS
16 02617 006254         LOOP
17
18 02620 006252 D19:    SETUP           ;CHECK PROPER CA REGISTER
19 02621 022145         LDA 0,TESTU     ;INCREMENT BY ALLOWING
20 02622 063033         DOC 0,DSKP      ;THE FIRST 2 DATA CHANNEL
21 02623 102000         ADC 0,0         ;CYCLES AT THE BEGINNING
22 02624 062033         DOB 0,DSKP      ;OF A WRITE.
23 02625 024062         LDA 1,KB7
24 02626 065133         DOAS 1,DSKP     ; WRITE !!
25 02627 006160         JSR @STALL
26 02630 062233         NIOC DSKP       ;STOP THE WRITE
27 02631 065433         DIB 1,DSKP      ;AC0=STARTING MEMORY ADDRESS
28 02632 152020         SUBZR 2,2       ;AC1=ACTUAL ENDING MEMORY
29 02633 131400         INC 2,2         ;     ADDRESS
30 02634 146414         SUB# 2,1,SZR    ;AC2=CORRECT ENDING MEMORY
31 02635 006253         EHALT           ;     ADDRESS
32 02636 006254         LOOP
33
34
35
```

```
10037 .MAIN
#1
#2 02637 006251 D2#1     SETP1          ;ATTEMPT A COMPLETE WRITE
#3 02640 024153          LDA 1,BUFF     ;(FIRST WRITE WITH NO OVFLO)
#4 02641 066033          DOB 1,DSKP     ;ONE SECTOR
#5 02642 020145          LDA 0,TESTU
#6 02643 024114          LDA 1,C17
#7 02644 123000          ADD 1,0        ;UNIT # + SECT CNT
#8 02645 063033          DOC 0,DSKP     ;SELECT UNIT
#9 02646 020062          LDA 0,KB7
10 02647 061133          DOAS 0,DSKP    ;WRITE !!
11 02650 006246          ITRWT          ;WAIT 100MS OR UNTIL INTERRUPT
12 02651 006253          EHALT          ;TIMEOUT, NO INTERRUPT
13 02652 006254          LOOP
14
15 02653 006251 D211     SETP1          ;ATTEMPT TO CLEAR "DP DONE"
16 02654 024153          LDA 1,BUFF     ;WITH (C) PULSE.
17 02655 066033          DOB 1,DSKP
18 02656 020145          LDA 0,TESTU
19 02657 024114          LDA 1,C17      ;SET "DP DONE" WITH A ONE
20 02660 123000          ADD 1,0        ;SECTOR WRITE.
21 02661 063033          DOC 0,DSKP     ;SELECT UNIT
22 02662 020062          LDA 0,KB7
23 02663 061133          DOAS 0,DSKP    ;WRITE !!
24 02664 152520          SUBZL 2,2
25 02665 006240          WAIT           ;WAIT 100MS OR UNTIL DONE
26 02666 063233          NIOC DSKP      ;ATTEMPT CLEAR
27 02667 063733          SKPDZ DSKP     ;(C) PULSE DOES NOT
28 02670 006253          EHALT          ;CLEAR "DP DONE" FF.
29 02671 006254          LOOP
30
31 02672 006251 D221     SETP1          ;ATTEMPT TO RESET
32 02673 024153          LDA 1,BUFF     ;"DP DONE" WITH "DP DATOA"
33 02674 066033          DOB 1,DSKP     ;AND "DATA 0".
34 02675 020145          LDA 0,TESTU
35 02676 024114          LDA 1,C17      ;SET "DP DONE" WITH A 1 SECT.
36 02677 123000          ADD 1,0        ;WRITE.
37 02700 063033          DOC 0,DSKP     ;SELECT UNIT
38 02701 020062          LDA 0,KB7
39 02702 061133          DOAS 0,DSKP    ;WRITE !!
40 02703 152520          SUBZL 2,2
41 02704 006240          WAIT           ;WAIT 100MS OR UNTIL DONE
42 02705 102520          SUBZR 0,0
43 02706 061033          DOA 0,DSKP     ;ATTEMPT RESET
44 02707 063733          SKPDZ DSKP     ;"DP DATOA" AND "DATA 0" DOES
45 02710 006253          EHALT          ;NOT CLEAR "DP DONE".
46 02711 006254          LOOP
```

```
10038 .MAIN
#1
#2 02712 006251 D23:     SETP1          ;CHECK BUSY-DONE
#3 02713 024153          LDA 1,BUFF     ;FOLLOWING A 1 SECTOR WRITE
#4 02714 066033          DOB 1,DSKP
#5 02715 020145          LDA 0,TESTU
#6 02716 024114          LDA 1,C17
#7 02717 123000          ADD 1,0
#8 02720 063033          DOC 0,DSKP     ;SELECT UNIT / 1 SECT XFER
#9 02721 020062          LDA 0,KB7
10 02722 061133          DOAS 0,DSKP    ;WRITE !!
11 02723 152520          SUBZL 2,2
12 02724 006240          WAIT           ;WAIT 100MS (OR UNTIL "DONE")
13 02725 063733          SKPDZ DSKP
14 02726 063533          SKPBZ DSKP     ;ERROR IF "DP DONE" = 0
15 02727 006253          EHALT          ;OR "DP BUSY" = 1
16 02730 006254          LOOP
17
18 02731 006251 D24:     SETP1          ;SEE IF A 1 SECTOR
19 02732 024153          LDA 1,BUFF     ;WRITE CAUSES "DATA LATE"
20 02733 066033          DOB 1,DSKP
21 02734 020145          LDA 0,TESTU
22 02735 024114          LDA 1,C17
23 02736 123000          ADD 1,0
24 02737 063033          DOC 0,DSKP     ;SELECT UNIT / 1 SECT XFER
25 02740 020062          LDA 0,KB7
26 02741 061133          DOAS 0,DSKP    ;WRITE !!
27 02742 152520          SUBZL 2,2
28 02743 006240          WAIT           ;WAIT 100MS, (OR UNTIL "DONE")
29 02744 064433          DIA 1,DSKP
30 02745 020053          LDA 0,KB14
31 02746 107414          AND# 0,1,SZR   ;"REQ1" DOES NOT CLEAR
32 02747 006253          EHALT          ;"DP FLAG".  RESULTS IN
33 02750 063254          LOOP           ;"DATA LATE" STATUS.
34
35 02751 006251 D25:     SETP1          ;CHECK ENDING MEMORY
36 02752 024153          LDA 1,BUFF     ;ADDRESS (CA REGISTER)
37 02753 063633          DOB 1,DSKP     ;FOLLOWING A 1 SECTOR WRITE
38 02754 020146          LDA 0,TESTU
39 02755 024114          LDA 1,C17
40 02756 123000          ADD 1,0
41 02757 063033          DOC 0,DSKP     ;SELECT UNIT / 1 SECT XFER
42 02760 020062          LDA 0,KB7
43 02761 061133          DOAS 0,DSKP    ;WRITE !!
44 02762 152520          SUBZL 2,2
45 02763 006240          WAIT           ;WAIT 100MS ( OR UNTIL "DONE" )
46 02764 065433          DIB 1,DSKP     ;AC1=ENDING MEMORY ADDRESS
47 02765 020153          LDA 0,BUFF     ;CORRECT ENDING ADDRESS =
48 02766 030214          LDA 2,C402     ;(BUFF)+400+2, IN (AC0)
49 02767 143002          ADD 2,0
50 02770 122414          SUBN 1,0,SZR   ;ENDING MEM ADDR ERROR.
51 02771 006253          EHALT          ;AC0=GOOD
52 02772 006254          LOOP           ;AC1=BAD
```

```
10039 .MAIN
#1
02 02773 006251 D26:    SETP1           ;CHECK FOR PROPER ENDING
03 02774 024153         LDA 1,BUFF      ;DISK ADDRESS FOLLOWING
04 02775 066033         DOB 1,DSKP      ;1 SECTOR WRITE ON HEAD 0
05 02776 020145         LDA 0,TESTU     ;SECTOR 0
06 02777 024114         LDA 1,C17       ;"INC SC" CAUSES "SC1-SC0"
07 03000 123000         ADD 1,0         ;TO GO FROM 1111 TO 0000
08 03001 063033         DOC 0,DSKP      ;AND SETS "S1".
09 03002 020062         LDA 0,KB7
10 03003 061133         DOAS 0,DSKP     ;WRITE !!
11 03004 152520         SUBZL 2,2
12 03005 006240         WAIT            ;WAIT 100MS (OR UNTIL "DONE")
13 03006 066433         DIC 1, DSKP     ;READ ENDING DISK ADDRESS
14 03007 030126         LDA 2,C037
15 03010 147400         AND 2,1         ;MASK OUT UNIT #
16 03011 020056         LDA 0,KB11      ;SECTOR = 1
17 03012 122414         SUB# 1,0,SZR
18 03013 006253         EHALT           ;ENDING DISK ADDRESS NOT
19 03014 006254         LOOP            ;HEAD=0, SECTOR=1, SC=0
20
21 03015 006251 D27:    SETP1           ;ATTEMPT A 2 SECTOR
22 03016 020153         LDA 0,BUFF      ;WRITE.
23 03017 062033         DOB 0,DSKP      ;FIRST ATTEMPT AT A WRITE
24 03020 020145         LDA 0,TESTU     ;LONGER THAN 1 SECTOR
25 03021 024167         LDA 1,C16
26 03022 123020         ADD 1,0
27 03023 053033         DOC 0,DSKP      ;SELECT UNIT / 2 SECTORS
28 03024 020062         LDA 0,KB7
29 03025 061133         DOAS 0,DSKP     ;WRITE !!
30 03026 152520         SUBZL 2,2
31 03027 006240         WAIT            ;WAIT 100MS (OR UNTIL "DONE")
32 03030 066433         DIC 1,DSKP      ;READ ENDING DISK ADDRESS
33 03031 020126         LDA 0,C037      ;MASK OFF UNIT #
34 03032 107400         AND 0,1
35 03033 020057         LDA 0,KB10
36 03034 122414         SUB# 1,0,SZR    ;ENDING DISK ADDRESS NOT
37 03035 006253         EHALT           ;HEAD=0, SECT=2, SC=0
38 03036 006254         LOOP            ;AC0=GOOD ADDR, AC1=BAD
39
40 03037 006251 D28:    SETP1           ;ATTEMPT A 2 SECTOR WRITE
41 03040 020153         LDA 0,BUFF
42 03041 062033         DOB 0,DSKP
43 03042 020145         LDA 0,TESTU
44 03043 024167         LDA 1,C16
45 03044 123020         ADD 1,0
46 03045 053033         DOC 0,DSKP      ;SELECT UNIT / 2 SECTORS
47 03046 020062         LDA 0,KB7
48 03047 061133         DOAS 0,DSKP     ;WRITE !!
49 03050 152520         SUBZL 2,2
50 03051 006240         WAIT            ;WAIT 100MS )OR UNTIL "DONE")
51 03052 064433         DIA 1,DSKP      ;READ STATUS
52 03053 063633         SKPDN DSKP      ;NO "DP DONE"
53 03054 006253         EHALT           ;FOLLOWING 2 SECTOR WRITE
54 03055 006254         LOOP            ;(AC1)=STATUS REG.
```

```
10040 .MAIN
#1
02 03056 006251 D29:    SETP1           ;INSURE THAT "INC HEAD"
03 03057 024153         LDA 0,BUFF      ;FUNCTIONS PROPERLY
04 03060 062033         DOB 0,DSKP      ;TRANSFER 7 SECTORS OR 13
05 03061 020145         LDA 0,TESTU     ;SECTORS DEPENDING UPON
06 03062 030143         LDA 2,DTYPE     ;THE DISK TYPE
07 03063 024164         LDA 1,C11
08 03064 153103         ADDL 2,2,SNC
09 03065 024112         LDA 1,C3
10 03066 123000         ADD 1,0
11 03067 063033         DOC 0,DSKP      ;LOAD UNIT # & # SECTORS
12 03070 020062         LDA 0,KB7
13 03071 061133         DOAS 0,DSKP     ;WRITE !!
14 03072 152520         SUBZL 2,2
15 03073 006240         WAIT            ;WAIT 100 MS (OR UNTIL "DONE")
16 03074 066433         DIC 1,DSKP      ;READ ENDING DISK ADDRESS
17 03075 030126         LDA 2,C037
18 03076 147400         AND 2,1         ;THROW AWAY UNIT #
19 03077 020215         LDA 0,C420      ;AC0=GOOD,    HEAD=1 SECT=1
20 03100 122414         SUB# 1,0,SZR    ;AC1=BAD
21 03101 006253         EHALT           ;CHECK "ADV HD", "INC HEAD"
22 03102 006254         LOOP
23
24 03103 006251 D30:    SETP1           ;ATTEMPT A 16 SECTOR WRITE
25 03104 020153         LDA 0,BUFF      ;VERIFY CORRECT ENDING
26 03105 062033         DOB 0,DSKP      ;DISK ADDRESS
27 03106 020145         LDA 0,TESTU
28 03107 063033         DOC 0,DSKP      ;UNIT # / 16 SECYORS
29 03110 020062         LDA 0,KB7
30 03111 061133         DOAS 0,DSKP     ;WRITE !!
31 03112 152520         SUBZL 2,2
32 03113 006240         WAIT            ;WAIT 100MS (OR UNTIL "DONE")
33 03114 066433         DIC 1,DSKP      ;READ ENDING DISK ADDRESS
34 03115 020060         LDA 0,KB9       ;SECTOR 4
35 03116 034063         LDA 3,KB6       ;HEAD 2
36 03117 030143         LDA 2,DTYPE
37 03120 153103         ADDL 2,2,SNC    ;SKIP IF 2311
38 03121 175220         MOVZR 3,3       ;HEAD 1
39 03122 163000         ADD 3,0
40 03123 034126         LDA 3,C037
41 03124 167400         AND 3,1
42 03125 122414         SUB# 1,0,SZR    ;AC1=ACTUAL ENDING DISK ADDRESS
43 03126 006253         EHALT           ;AC0=CORRECT ENDING DISK ADDRESS
44 03127 006254         LOOP            ) 16 SECTOR TRANSFER.
```

```
;M041 .MAIN
01
02 03130 006251 D31:   SETP1           ;ATTEMPT A 16 SECTOR WRITE
03 03131 020153        LDA 0,BUFF      ;AND VERIFY CORRECT ENDING
04 03132 062733        DOB 0,DSKP      ;MEMORY ADDRESS
05 03133 020145        LDA 0,TESTU
06 03134 063033        DOC 0,DSKP      ;UNIT # / 16 SECTORS
07 03135 020062        LDA 0,KB7
08 03136 061133        DOAS 0,DSKP     ;WRITE !!
09 03137 152520        SUBZL 2,2
10 03140 006240        WAIT            ;WAIT 100MS (OR UNTIL "DONE")
11 03141 020153        LDA 0,BUFF      ;STARTING ADDRESS
12 03142 024053        LDA 1,KB14      ; TWO
13 03143 032060        LDA 2,KB3       ; 10,000
14 03144 123000        ADD 1,0
15 03145 143000        ADD 2,0         ;ENDING MEMORY ADDRESS
16 03146 065433        DIB 1,DSKP      ;ERROR FOLLOWING A 16 SECTOR
17 03147 122414        SUB# 1,0,SZR    ;WRITE
18 03150 006253        EHALT           ;AC0=GOOD
19 03151 006254        LOOP            ;AC1=BAD
20
21 03152 006251 D32:   SETP1           ;CAUSE "END OF CYLINDER"
22 03153 020143        LDA 0,DTYPE     ;TO OCCUR DURING A
23 03154 030427        LDA 2,DAD1-1    ;2 SECTOR WRITE
24 03155 101112        MOVL# 0,0,SZC
25 03156 000404        JMP .+4         ;CARTRIDGE DISK
26 03157 151400        INC 2,2
27 03160 101202        MOVR 0,0,SZC    ;SKIP IF 2311 DISK PACK
28 03161 151400        INC 2,2         ;2314 DISK PACK
29 03162 021000        LDA 0,0,2       ;GET PROPER DISK ADDRESS TO
30 03163 024145        LDA 1,TESTU     ;CAUSE "EOC"
31 03164 123000        ADD 1,0         ;ADD IN UNIT #
32 03165 063833        DOC 0,DSKP      ;FINALLY
33 03166 020153        LDA 0,BUFF
34 03167 062833        DOB 0,DSKP      ;LOAD MEM ADDR.
35 03170 020062        LDA 0,KB7
36 03171 061133        DOAS 0,DSKP     ;WRITE !!
37 03172 152520        SUBZL 2,2
38 03173 006240        WAIT            ;WAIT 100MS (OR UNTIL "DONE")
39 03174 064433        DIA 1,DSKP      ;READ STATUS
40 03175 020412        LDA 0,DAD4
41 03176 107400        AND 0,1
42 03177 106414        SUB# 0,1,SZR    ;NO "EOC" OR "ERR" OR "DP DONE"
43 03200 006253        EHALT           ;FOLLOWING WRITE OVER END CYL
44 03201 006254        LOOP            ;AC0=EXPECTED ENDING STATUS
45 03202 000407        JMP D32A        ;AC1=ENDING STATUS
46
47 03203 003204        .+1
48 03204 000676 DAD1:  676             ;CART   HEAD 1  SECT 13 2 SECTORS
49 03205 004536        4536            ;2311   HEAD 9  SECT 5  2 SECTORS
50 03206 011676        11676           ;2314   HEAD 23 SECT 13 2 SECTORS
51 03207 100021 DAD4:  100021          ;"DP DONE","EOC","ERR",STATUS MASK.
52 03210 000020 EOCNT: 20
```

```
;M042 .MAIN
01
02 03211 006251 D32A:  SETP1           ;DRIVE DISK ADAPTER
03 03212 020143        LDA 0,DTYPE
04 03213 101113        MOVL# 0,0,SNC
05 03214 000423        JMP D32C
06 03215 020056        LDA 0,KB11      ;INTO END OF CYLINDER TO
07 03216 040772        STA 0,EOCNT     ;CHECK INC HD FLOP IN
08 03217 020145        LDA 0,TESTU     ;ADAPTER LOGIC. EOC
09 03220 063033        DOC 0,DSKP      ;FOR CARTRIDGE OCCURS ON
10 03221 020153 D32B:  LDA 0,BUFF      ;2ND WRITE CYCLE.
11 03222 062033        DOB 0,DSKP      ;
12 03223 020062        LDA 0,KB7       ;
13 03224 061133        DOAS 0,DSKP     ;
14 03225 152520        SUBZL 2,2       ;EOCNT
15 03226 006240        WAIT            ;16 SECTOR WRITE CYCLE
16 03227 064433        DIA 1,DSKP      ;COUNTER IS LOADED WITH
17 03230 020757        LDA 0,DAD4      ;MAX. NUMBER OF LOOPS
18 03231 107400        AND 0,1         ;REQUIRED TO ACHIEVE
19 03232 106415        SUB# 0,1,SNR    ;AN EOC AFTER A START
20 03233 000404        JMP .+4         ;FROM HD0, SECTOR 0.
21 03234 014754        DSZ EOCNT       ;KB11(C) OCTAL 26.
22 03235 000754        JMP D32B        ;NO EOC & MORE LOOPS
23 03236 006253        EHALT           ;NO EOC. CK INC HD, HEAD,&
24 03237 006254 D32C:  LOOP            ;EOC FLOP IN ADAPTER LOGIC.
```

```
10043 .MAIN
#1
#2 03240 006251 D33:    SETP1           ;ATTEMPT A READ
#3 03241 020145         LDA 0,TESTU     ;( FIRST READ !!!!! )
#4 03242 024114         LDA 1,C17
#5 03243 123000         ADD 1,0
#6 03244 063033         DOC 0,DSKP      ;UNIT # / 1 SECTOR XFER
#7 03245 020067         LDA 0,KB2
#8 03246 062033         DOB 0,DSKP      ;CA = 20000
#9 03247 100400         SUB 0,0
10 03250 061133         DOAS 0,DSKP     ;READ !!
11 03251 152520         SUBZL 2,2
12 03252 006240         WAIT            ;WAIT 10MS (OR UNTIL "DONE")
13 03253 064433         DIA 1,DSKP      ;READ STATUS
14 03254 125113         MOVL# 1,1,SNC   ; DONE ?
15 03255 006253         EHALT           ;NO "OP DONE" ON READ
16 03256 006254         LOOP            ;AC1=BAD ENDING STATUS
17
18 03257 006251 D34:    SETP1           ;ATTEMPT A READ
19 03260 020146         LDA 0,TESTU     ; 1 SECTOR
20 03261 024114         LDA 1,C17
21 03262 123000         ADD 1,0
22 03263 063033         DOC 0,DSKP      ;UNIT # / 1 SECTOR XFER
23 03264 020067         LDA 0,KB2
24 03265 062033         DOB 0,DSKP      ;CA = 20000
25 03266 100400         SUB 0,0
26 03267 061133         DOAS 0,DSKP     ;READ !!
27 03270 152520         SUBZL 2,2
28 03271 006240         WAIT            ;WAIT 10MS (OR UNTIL "DONE")
29 03272 064433         DIA 1,DSKP      ;READ STATUS
30 03273 020053         LDA 0,KB14      ;"DATA LATE" STATUS ON
31 03274 123414         ANDZ# 1,0,SZR   ;A 1 SECT READ
32 03275 006253         EHALT           ;AC1=BAD ENDING STATUS
33 03276 006254         LOOP
34
35 03277 006251 D35:    SETP1           ;ATTEMPT A READ
36 03300 020145         LDA 0,TESTU     ; 1 SECTOR READ
37 03301 024114         LDA 1,C17
38 03302 123000         ADD 1,0
39 03303 063033         DOC 0,DSKP      ; UNIT # / 1 SECT XFER
40 03304 020067         LDA 0,KB2
41 03305 062033         DOB 0,DSKP      ;CA = 20000
42 03306 100400         SUB 0,0
43 03307 061133         DOAS 0,DSKP     ;READ !!
44 03310 152520         SUBZL 2,2
45 03311 006240         WAIT            ;WAIT 10MS (OR UNTIL "DONE")
46 03312 065433         DIB 1,DSKP      ;READ STATUS
47 03313 020224         LDA 0,C2004     ;ENDING MEMORY ADDRESS
48 03314 122414         SUBZ# 1,0,SZR   ;IS WRONG FOLLWOING 1 SECT READ
49 03315 006253         EHALT           ;AC0=GOOD
50 03316 006254         LOOP            ;AC1=BAD
```

```
10044 .MAIN
#1
#2 03317 006251 D36:    SETP1           ;ATTEMPT A READ
#3 03320 020145         LDA 0,TESTU     ; 1 SECTOR READ
#4 03321 024114         LDA 1,C17
#5 03322 123000         ADD 1,0
#6 03323 063033         DOC 0,DSKP      ;UNIT # / 1 SECT XFER
#7 03324 020067         LDA 0,KB2
#8 03325 062033         DOB 0,DSKP      ;CA = 20000
#9 03326 060133         NIOS DSKP       ;READ !!
10 03327 152520         SUBZL 2,2
11 03330 006240         WAIT            ;WAIT 10MS (OR UNTIL "DONE")
12 03331 020205         LDA 0,C70       ;AC1=STATUS
13 03332 123414         ANDZ# 1,0,SZR   ;ERROR STATUS FOLLOWING A
14 03333 006253         EHALT           ;READ, "SEEK ER", OR "ADDRESS/
15 03334 006254         LOOP            ;UNSAFE"
16
17 03335 020143 D37:    LDA 0,DTYPE     ;IF CARTRIDGE DISK
18 03336 101102         MOVL 0,0,SZC    ;DON'T ATTEMPT A FORMAT
19 03337 000416         JMP E1          ;MODE READ
20 03340 006251         SETP1           ;ATTEMPT A READ
21 03341 020145         LDA 0,TESTU     ;IN THE FORMAT MODE
22 03342 024067         LDA 1,KB2       ;(FIRST USE OF FORMAT MODE)
23 03343 123000         ADD 1,0
24 03344 063033         DOC 0,DSKP      ;UNIT # FORMAT MODE BIT
25 03345 020067         LDA 0,KB2
26 03346 062033         DOB 0,DSKP      ;CA = 20000
27 03347 060133         NIOS DSKP       ;READ!!
28 03350 152520         SUBZL 2,2
29 03351 006240         WAIT            ;WAIT 10MS (OR UNTIL "DONE")
30 03352 125113         MOVL# 1,1,SNC   ;AC1=STATUS
31 03353 006253         EHALT           ;NO "OP DONE" FOLLOWING FMT READ
32 03354 006254         LOOP            ;CHECK "F DONE","R/W DONE"
33
34
```

```
10045 .MAIN

03355 006250 E1:     JSR @ISET          ;DO SEEK/WRITE/READ
03356 006272         DOSEK              ;CHECK DATA
03357 000000         0                  ;SEEK CYLINDER ZERO
03360 006253         EHALT              ;ERROR DURING SEEK, AC1=STATUS
03361 000422         JMP E1F            ;SKIP REMAINDER OF TEST

03362 006266         GENDAT             ;GENERATE DATA
03363 005176         ZEROS              ;DATA=ALL ZEROS
03364 006530         PRGEND             ;DATA BUFFER ADDRESS

03365 006270         WRITE              ;DO THE WRITE
03366 006530         PRGEND             ;DATA BUFFER ADDRESS
03367 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
03370 006253         EHALT              ;ERROR IN WRITE, AC1=STATUS
03371 000412         JMP E1E            ;SKIP REMAINDER OF TEST

03372 006267         READ               ;READ THE DATA
03373 007230         PRGEND+400         ;DATA BUFFER ADDRESS
03374 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
03375 006253         EHALT              ;ERROR IN READ, AC1=STATUS
03376 000405         JMP E1E

03377 006271         CHECK              ;COMPARE DATA BUFFERS
03400 006530         PRGEND             ;GOOD BUFFER
03401 007230         PRGEND+400         ;QUESTIONABLE BUFFER
03402 006253         EHALT              ;ERROR, AC0=GOOD WORD
03403 006254 E1E:    LOOP               ;AC1=BAD WORD

03404 006250 E2:     JSR @ISET          ;DO SEEK/WRITE/READ
03405 006272         DOSEK              ;CHECK DATA
03406 000020         0                  ;SEEK CYLINDER ZERO
03407 006253         EHALT              ;ERROR DURING SEEK, AC1=STATUS
03410 000422         JMP E2E            ;SKIP REMAINDER OF TEST

03411 006266         GENDAT             ;GENERATE DATA
03412 005175         ONES               ;DATA=ALL ONES
03413 006530         PRGEND             ;DATA BUFFER ADDRESS

03414 006270         WRITE              ;DO THE WRITE
03415 006530         PRGEND             ;DATA BUFFER ADDRESS
03416 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
03417 006253         EHALT              ;ERROR IN WRITE, AC1=STATUS
03420 000412         JMP E2E            ;SKIP REMAINDER OF TEST

03421 006267         READ               ;READ THE DATA
03422 007230         PRGEND+400         ;DATA BUFFER ADDRESS
03423 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
03424 005253         EHALT              ;ERROR IN READ, AC1=STATUS
03425 000405         JMP E2F

03426 006271         CHECK              ;COMPARE DATA BUFFERS
03427 006530         PRGEND             ;GOOD BUFFER
03430 007230         PRGEND+400         ;QUESTIONABLE BUFFER
03431 006253         EHALT              ;ERROR, AC0=GOOD WORD
03432 006254 E2E:    LOOP               ;AC1=BAD WORD
```

```
10046 .MAIN

03433 006250 E3:     JSR @ISET          ;DO SEEK/WRITE/READ
03434 006272         DOSEK              ;CHECK DATA
03435 000000         0                  ;SEEK CYLINDER ZERO
03436 006253         EHALT              ;ERROR DURING SEEK, AC1=STATUS
03437 000422         JMP E3E            ;SKIP REMAINDER OF TEST

03440 006266         GENDAT             ;GENERATE DATA
03441 005214         NUMSEQ             ;DATA=NUMBERS 0-377
03442 006530         PRGEND             ;DATA BUFFER ADDRESS

03443 006270         WRITE              ;DO THE WRITE
03444 006530         PRGEND             ;DATA BUFFER ADDRESS
03445 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
03446 006253         EHALT              ;ERROR IN WRITE, AC1=STATUS
03447 000412         JMP E3E            ;SKIP REMAINDER OF TEST

03450 006267         READ               ;READ THE DATA
03451 007230         PRGEND+400         ;DATA BUFFER ADDRESS
03452 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
03453 006253         EHALT              ;ERROR IN READ, AC1=STATUS
03454 000405         JMP E3E

03455 006271         CHECK              ;COMPARE DATA BUFFERS
03456 006530         PRGEND             ;GOOD BUFFER
03457 007230         PRGEND+400         ;QUESTIONABLE BUFFER
03460 006253         EHALT              ;ERROR, AC0=GOOD WORD
03461 006254 E3E:    LOOP               ;AC1=BAD WORD

03462 006250 E4:     JSR @ISET          ;DO SEEK/WRITE/READ
03463 006272         DOSEK              ;CHECK DATA
03464 000000         0                  ;SEEK CYLINDER ZERO
03465 006253         EHALT              ;ERROR DURING SEEK, AC1=STATUS
03466 000422         JMP E4E            ;SKIP REMAINDER OF TEST

03467 006266         GENDAT             ;GENERATE DATA
03470 005204         ALT1               ;DATA PATTERN = 101010 ETC.
03471 006530         PRGEND             ;DATA BUFFER ADDRESS

03472 006270         WRITE              ;DO THE WRITE
03473 006530         PRGEND             ;DATA BUFFER ADDRESS
03474 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
03475 006253         EHALT              ;ERROR IN WRITE, AC1=STATUS
03476 000412         JMP E4E            ;SKIP REMAINDER OF TEST

03477 006267         READ               ;READ THE DATA
03500 007230         PRGEND+400         ;DATA BUFFER ADDRESS
03501 000017         17                 ;HEAD=0, SECT=0, 1 SECTOR
03502 006253         EHALT              ;ERROR IN READ, AC1=STATUS
03503 000405         JMP E4E

03504 006271         CHECK              ;COMPARE DATA BUFFERS
03505 006530         PRGEND             ;GOOD BUFFER
03506 007230         PRGEND+400         ;QUESTIONABLE BUFFER
03507 006253         EHALT              ;ERROR, AC0=GOOD WORD
03510 006254 E4E:    LOOP               ;AC1=BAD WORD
```

```
!0047 .MAIN
01
02 03511 000625R E5:    JSR @ISET       !DO SEEK/WRITE/READ
03 03512 006272         DOSEK           !CHECK DATA
04 03513 000000         0               !SEEK CYLINDER ZERO
05 03514 006253         EHALT           !ERROR DURING SEEK, AC1=STATUS
06 03515 000422         JMP E5F         !SKIP REMAINDER OF TEST
07
08 03516 006266         GENDAT          !GENERATE DATA
09 03517 005200         ALT0            !DATA PATTERN = 0101010 ETC.
10 03520 006630         PRGEND          !DATA BUFFER ADDRESS
11
12 03521 006270         WRITE           !DO THE WRITE
13 03522 006630         PRGEND          !DATA BUFFER ADDRESS
14 03523 000017         17              !HEAD=0, SECT=0, 1 SECTOR
15 03524 006253         EHALT           !ERROR IN WRITE, AC1=STATUS
16 03525 000412         JMP E5E         !SKIP REMAINDER OF TEST
17
18 03526 006267         READ            !READ THE DATA
19 03527 007230         PRGEND+400      !DATA BUFFER ADDRESS
20 03530 000017         17              !HEAD=0, SECT=0, 1 SECTOR
21 03531 006253         EHALT           !ERROR IN READ, AC1=STATUS
22 03532 000405         JMP E5E
23
24 03533 006271         CHECK           !COMPARE DATA BUFFERS
25 03534 006630         PRGEND          !GOOD BUFFER
26 03535 007230         PRGEND+400      !QUESTIONABLE BUFFER
27 03536 006253         EHALT           !ERROR, AC0=GOOD WORD
28 03537 000254 E5E:    LOOP            !AC1=BAD WORD
```

```
!0048 .MAIN
01
02 03540 002135         LDA 0,RANDOM
03 03541 004136         STA 0,RELRAN
04 03542 000625R E6:    JSR @ISET       !DO SEEK/WRITE/READ
05 03543 006272         DOSEK           !CHECK DATA
06 03544 000000         0               !SEEK CYLINDER ZERO
07 03545 006253         EHALT           !ERROR DURING SEEK, AC1=STATUS
08 03546 000424         JMP E6E         !SKIP REMAINDER OF TEST
09
10 03547 002136         LDA 0,RELRAN
11 03550 040135         STA 0,RANDOM
12 03551 005266         GENDAT          !GENERATE DATA
13 03552 005012         RAN             !DATA = RANDOM
14 03553 006630         PRGEND          !DATA BUFFER ADDRESS
15
16 03554 006270         WRITE           !DO THE WRITE
17 03555 006630         PRGEND          !DATA BUFFER ADDRESS
18 03556 000017         17              !HEAD=0, SECT=0, 1 SECTOR
19 03557 006253         EHALT           !ERROR IN WRITE, AC1=STATUS
20 03560 000412         JMP E6E         !SKIP REMAINDER OF TEST
21
22 03561 006267         READ            !READ THE DATA
23 03562 007230         PRGEND+400      !DATA BUFFER ADDRESS
24 03563 000017         17              !HEAD=0, SECT=0, 1 SECTOR
25 03564 006253         EHALT           !ERROR IN READ, AC1=STATUS
26 03565 000405         JMP E6E
27
28 03566 006271         CHECK           !COMPARE DATA BUFFERS
29 03567 006630         PRGEND          !GOOD BUFFER
30 03570 007230         PRGEND+400      !QUESTIONABLE BUFFER
31 03571 006253         EHALT           !ERROR, AC0=GOOD WORD
32 03572 000254 E6E:    LOOP            !AC1=BAD WORD
```

```
10049 .MAIN
01
02                ;DISCUSSION OF TESTS E7/E8 AND E9/E10
03
04                ;       AT THIS POINT IN TESTING IT HAS BEEN DETERMINED
05                ;THAT READING AND WRITING CAN BE PERFORMED CORRECTLY.  THE
06                ;NEXT FOUR TESTS ARE A CYLINDER ADDRESS CHECK.  CYLINDERS
07                ;ARE FIRST SELECTED IN ORDER (0-312) AND DATA EQUAL TO THE
08                ;CYLINDER NUMBER IS WRITTEN (TEST E7) ON HEAD=0
09                ;SECTOR=0 OF EACH.  THE FOLLOWING TEST THEN READS THIS DATA
10                ;BACK IN THE SAME SEQUENCE IN WHICH IT WAS WRITTEN.  IF ONE
11                ;CYLINDER IS WRITTEN ON MORE THAN ONCE DUE TO A (ALWAYS 0 OR
12                ;ALWAYS 1) CYLINDER ADDRESS BIT LINE AN ERROR WILL SHOW UP
13                ;IN THE DATA COMPARE CHECK.
14
15                ; I.E.  IF "CYL 4" IS ALWAYS A ZERO THEN AN ATTEMPT
16                ;       TO SEEK TO CYLINDER 4 ACTUALLY SELECTS
17                ;       CYLINDER 0.  DATA WORDS OF "4" ARE WRITTEN
18                ;       THERE AND WHEN THE DATA AT CYLINDER 0 IS
19                ;       LATER READ AND CHECKED IT WILL BE "4" WHEN
20                ;       IT SHOULD BE "0".
21
22                ;TESTS E7&E8 CHECK ALL CYLINDERS IN SEQUENCE 0-312
23                ;AND TESTS E9&E10 CHECK THE CYLINDERS IN ORDER
24                ;FROM 312 TO 0, USING THE COMPLEMENT OF THE CYLINDER
25                ;NUMBER AS THE DATA WORDS.
```

```
10050 .MAIN
01
02 03573 102400           SUB 0,0          ;CYL. ADDRESSING CHECK. SEE DISCUSSION
03 03574 040443           STA 0,E7.1       ;PRECEDING E7. WRITE ON HEAD=0, SECTOR=0
04 03575 006250 E7:       JSR @ISET        ;OF EACH CYLINDER.  IN EACH SECTOR DATA
05                                         ;EQUALS THE CYL #.  WRITE IN ASCENDING
06                                         ;ORDER FROM CYL 0 TO 312.
07 03576 006272           DOSEK            ;SEEK
08 03577 000000 E7.1:     0                ;CYLINDER NUMBER (IT CHANGES)
09 03600 006253           EHALT            ;ERROR DURING SEEK, AC1=STATUS
10 03601 000411           JMP E7E          ;SKIP TO END OF TEST
11 03602 006266           GENDAT           ;GENERATE DATA
12 03603 005212           CYLN             ;ADDRESS OF DATA GENERATOR
13 03604 006630           PRGEND           ;DATA BUFFER ADDRESS
14 03605 006270           WRITE            ;WRITE !
15 03606 006630           PRGEND           ;DATA BUFFER ADDRESS
16 03607 000017           17               ;DISK ADDRESS
17 03610 006253           EHALT            ;ERROR DURING WRITE, AC1=STATUS
18 03611 000401           JMP .+1
19 03612 006254 E7E:      LOOP
20 03613 010764           ISZ E7.1         ;(E7.1)=CYL #
21 03614 020763           LDA 0,E7.1       ;DO IT ONCE FOR
22 03615 024212           LDA 1,C312       ;EACH CYLINDER
23 03616 122427           SUBZ 1,0,SBN
24 03617 000736           JMP E7
25
26
27
28
29 03620 102400           SUB 0,0          ;CYLINDER ADDRESS CHECK.  SEE DISCUSSION
30 03621 040403           STA 0,E8.1       ;PREVIOUS TO TEST E7.  READ FIRST SECTOR
31 03622 006250 E8:       JSR @ISET        ;OF EACH CYLINDER AND CHECK FOR PROPER
32                                         ;DATA.  IN EACH CASE DATA SHOULD EQUAL
33                                         ;THE CYLINDER #.
34 03623 006272           DOSEK            ;SEEK
35 03624 000000 E8.1:     0                ;CYLINDER NUMBER (IT CHANGES)
36 03625 006253           EHALT            ;ERROR DURING SEEK, AC1=STATUS
37 03626 000412           JMP E8E
38 03627 006267           READ             ;READ
39 03630 006630           PRGEND           ;DATA BUFFER ADDRESS
40 03631 000017           17               ;DISK ADDRESS
41 03632 006253           EHALT            ;ERROR DURING READ, AC1=STATUS
42 03633 000403           JMP E8E          ;SKIP TO END OF TEST
43 03634 026774           LDA 1,@E8.2      ;FIRST WORD IN DATA BUFFER
44 03635 020767           LDA 0,E8.1       ;SHOULD = CYLINDER #
45 03636 122414           SUB# 1,0,SZR     ;AC0=GOOD
46 03637 006253           EHALT            ;AC1=BAD
47 03640 006254 E8E:      LOOP             ;READ THE TEST DESCRIPTION
48 03641 010753           ISZ E8.1         ;REPEAT THE TEST FOR
49 03642 020752           LDA 0,E8.1       ;EACH CYLINDER
50 03643 024212           LDA 1,C312
51 03644 122427           SUBZ 1,0,SBN
52 03645 000755           JMP E8
```

```
10051 .MAIN
01
02 03646 020212          LDA 0,C312      ;CYL. ADDRESSING CHECK. SEE DISCUSSION
03 03647 040403          STA 0,E9.1      ;PRECEDING TEST E7.  WRITE ON HEAD-0
04 03650 006250 E9:      JSR @ISET       ;SECTOR-0, OF EACH CYLINDER.  IN
05                                       ;EACH THE DATA WORDS EQUAL THE COMP.
06                                       ;OF THE CYL #.  WRITE IN DESCENDING
07                                       ;ORDER FROM CYL 312 TO 0.
08 03651 006272          DOSEK           ;SEEK
09 03652 000000 E9.1:    0               ;CYLINDER NUMBER (IT CHANGES)
10 03653 006253          EHALT           ;ERROR DURING SEEK, AC1=STATUS
11 03654 000411          JMP E9F         ;SKIP TO END OF TEST
12 03655 006266          GENDAT          ;GENERATE DATA
13 03656 005210          CYLNC           ;ADDRESS OF DATA GENERATOR
14 03657 006030          PRGEND          ;DATA BUFFER ADDRESS
15 03660 006270          WRITE           ;WRITE 1
16 03661 006030          PRGEND          ;DATA BUFFER ADDRESS
17 03662 000017          17              ;DISK ADDRESS
18 03663 006253          EHALT           ;ERROR DURING WRITE, AC1=STATUS
19 03664 000401          JMP .+1
20 03665 006254 E9E:     LOOP
21 03666 014764          DSZ E9.1        ;(E9.1)=CYL #
22 03667 000401          JMP .+1
23 03670 020752          LDA 0,E9.1      ;DO IT ONCE FOR
24 03671 101103          MOVL 0,0,SNC    ;EACH CYLINDER
25 03672 000756          JMP E9
26
27
28
29
30 03673 020212          LDA 0,C312      ;CYLINDER ADDRESS CHECK.  SEE DISCUSSION
31 03674 040403          STA 0,E10.1     ;PREVIOUS TO TEST E7.  READ FIRST SECTOR
32 03675 006250 E10:     JSR @ISET       ;OF EACH CYLINDER AND CHECK FOR PROPER
33                                       ;DATA.  IN EACH CASE DATA SHOULD EQUAL
34                                       ;THE COMPLEMENT OF THE CYL #
35 03676 006272          DOSEK           ;SEEK
36 03677 000000 E10.1:   0               ;CYLINDER NUMBER (IT CHANGES)
37 03700 006253          EHALT           ;ERROR DURING SEEK, AC1=STATUS
38 03701 000413          JMP E10E
39 03702 006267          READ            ;READ
40 03703 006030 E10.2:   PRGEND          ;DATA BUFFER ADDRESS
41 03704 000017          17              ;DISK ADDRESS
42 03705 006253          EHALT           ;ERROR DURING READ, AC1=STATUS
43 03706 000406          JMP E10E        ;SKIP TO END OF TEST
44 03707 026774          LDA 1,@E10.2    ;FIRST WORD IN DATA BUFFER
45 03710 020767          LDA 0,E10.1     ;SHOULD = CYLINDER #
46 03711 100900          COM 0,0
47 03712 122414          SUB# 1,0,SZR    ;AC2=GOOD
48 03713 006253          EHALT           ;AC1=BAD
49 03714 006254 E10E:    LOOP            ;READ THE TEST DESCRIPTION
50 03715 014762          DSZ E10.1       ;REPEAT THE TEST FOR
51 03716 000401          JMP .+1
52 03717 020750          LDA 0,E10.1     ;EACH CYLINDER
53 03720 101103          MOVL 0,0,SNC
54 03721 000754          JMP E10
55
56
```

```
10052 .MAIN
01
02               ;DISCUSSION OF THE SECTOR ADDRESSING CHECK.
03               ;(TESTS E11/E12 AND E13/E14)
04
05               ;USING CYLINDER-0, HEAD-0 TEST E11 WRITES ON
06               ;EACH SUCCESSIVE SECTOR INDIVIDUALLY FROM 0-5,
07               ;OR 0-11.  IN EACH CASE EACH DATA WORD EQUALS
08               ;THE SECTOR NUMBER.  TEST E12 READS THE INDIV-
09               ;IDUAL SECTORS BACK IN THE SAME ORDER AND
10               ;CHECKS THE DATA.  TESTS E13/E14 PERFORM THE SAME
11               ;TASK EXCEPT THAT THE SECTOR SEQUENCE IS IN THE
12               ;REVERSE ORDER AND THE DATA WRITTEN EQUALS THE
13               ;COMPLEMENT OF THE SECTOR NUMBER.
14
15               ;THIS TEST IS DESIGNED TO CATCH ERRONEOUS SECTOR
16               ;SELECTION ERRORS.
17
18               ;  I.E. IF "SC2" IS ALWAYS AT GROUND WHEN SECTOR
19               ;       2 IS SELECTED, SECTOR 0 WILL ACTUALLY
20               ;       BE CHOSEN AND "2'S" WILL BE WRITTEN THERE.
21               ;       IN THE SUBSEQUENT READ A DATA COMPARE
22               ;       ERROR WILL RESULT WITH THE GOOD="0",
23               ;       AND THE BAD="2".
24
25 03722 102400          SUB 0,0         ;SEE DISCUSSION PRECEDING THIS TEST.
26 03723 040152          STA 0,SECT      ;WRITE ON CYL-0, HEAD-0, ONCE
27 03724 020114          LDA 0,C17       ;FOR EACH SECTOR.  IN EACH
28 03725 040413          STA 0,E11.1     ;CASE THE DATA WORDS EQUAL
29 03726 006250 E11:     JSR @ISET       ;THE SECTOR NUMBER
30 03727 006272          DOSEK           ;SEEK 1
31 03730 000000          0               ;CYLINDER 0
32 03731 006253          EHALT           ;ERROR IN SEEK, AC1=STATUS
33 03732 000411          JMP E11E        ;SKIP TO END OF TEST
34 03733 006266          GENDAT          ;GENERATE DATA
35 03734 005233          SECTN           ;ADDRESS OF DATA GEN ROUT.
36 03735 006030          PRGEND          ;ADDRESS OF DATA BUFFER
37 03736 006270          WRITE           ;WRITE 1
38 03737 006030          PRGEND          ;DATA BUFFER ADDRESS
39 03740 000017 E11.1:   17              ;DISK ADDRESS (IT CHANGES)
40 03741 006253          EHALT           ;ERROR IN WRITE, AC1=STATUS
41 03742 000401          JMP .+1
42 03743 006254 E11E:    LOOP
43
44 03744 010152          ISZ SECT        ;INCREMENT THE SECTOR
45 03745 020773          LDA 0,E11.1     ;NUMBER
46 03746 024056          LDA 1,K811
47 03747 123000          ADD 1,0
48 03750 040770          STA 0,E11.1
49 03751 024143          LDA 1,DTYPE     ;SEE IF DONE
50 03752 030210          LDA 2,C157
51 03753 127103          ADDL 1,1,SNC
52 03754 030213          LDA 2,C317
53 03755 112404          SUB 0,2,SZR
54 03756 000750          JMP E11         ;DO ANOTHER
```

```
!0053 .MAIN
#1
#2 03757 102400          SUB #,#          ;SEE THE DISCUSSION PRECEDING
#3 03760 044152          STA #,SECT       ;TEST E11.  READ ONCE FROM
#4 03761 020114          LDA #,C17        ;EACH SECTOR AT CYL=#, HEAD=#.
#5 03762 044417          STA #,E12,2      ;CHECK DATA IN EACH CASE.
#6 03763 006250 E12:     JSR @ISET        ;DATA WORDS=SECTOR #.
#7 03764 006272          DOSEK            ;SEEK
#8 03765 000000          #                ;CYLINDER #
#9 03766 006253          EHALT            ;ERROR IN SEEK, AC1=STATUS
10 03767 000412          JMP E12E         ;SKIP TO END OF TEST
11 03770 006267          READ             ;READ !
12 03771 006630 E12,1:   PRGEND           ;DATA BUFFER ADDRESS
13 03772 000017 E12,2:   17               ;DISK ADDRESS (IT CHANGES)
14 03773 006253          EHALT            ;ERROR IN READ, AC1=STATUS
15 03774 000405          JMP E12E         ;SKIP TO END OF TEST
16 03775 020152          LDA #,SECT       ;GET SECT #
17 03776 026773          LDA 1,@E12,1     ;GET A WORD READ
18 03777 122414          SUB# 1,0,SZR     ;DATA ERROR, SEE ABOVE DESCRIPTION
19 04000 006253          EHALT            ;AC0=GOOD WORD
20 04001 006254 E12E:    LOOP             ;AC1=BAD
21
22 04002 010152          ISZ SECT         ;INCREMENT THE SECTOR
23 04003 020767          LDA #,E12,2      ;NUMBER
24 04004 024056          LDA 1,KB11
25 04005 123400          ADD 1,0
26 04006 040764          STA #,E12,2
27 04007 024143          LDA 1,DTYPE      ;SEE IF DONE
28 04010 030214          LDA 2,C157
29 04011 127103          ADDL 1,1,SNC
30 04012 030213          LDA 2,C317
31 04013 112404          SUB #,2,SZR
32 04014 000747          JMP E12          ;GO AGAIN
```

```
!0054 .MAIN
#1
#2 04015 020143          LDA #,DTYPE      ;SEE DISCUSSION PRECEDING
#3 04016 024207          LDA 1,C137       ;TEST E11.  WRITE ON CYL=#,
#4 04017 103103          ADDL #,#,SNC     ;HEAD=#, ONCE FOR EACH SECTOR.
#5 04020 024211          LDA 1,C277       ;IN EACH CASE THE DATA EQUALS
#6 04021 044420          STA 1,E13,1      ;THE COMPLEMENT OF THE SECTOR
#7 04022 125220          MOVZR 1,1        ;NUMBER
#8 04023 125220          MOVZR 1,1
#9 04024 125220          MOVZR 1,1
10 04025 125220          MOVZR 1,1
11 04026 044152          STA 1,SECT       ;BEGINNING SECTOR
12 04027 006250 E13:     JSR @ISET        ;SETUP
13 04030 006272          DOSEK            ;SEEK !
14 04031 000000          #                ;CYLINDER #
15 04032 006253          EHALT            ;ERROR IN SEEK, AC1=STATUS
16 04033 000411          JMP E13E         ;SKIP TO END OF TEST
17 04034 006266          GENDAT           ;GENERATE DATA
18 04035 005231          SETNC            ;ADDRESS OF DATA GENERATOR
19 04036 006630          PRGEND           ;ADDRESS OF DATA BUFFER
20 04037 006270          WRITE            ;WRITE !
21 04040 006630          PRGEND           ;ADDRESS OF DATA BUFFER
22 04041 000017 E13,1:   17               ;DISK ADDRESS (IT CHANGES)
23 04042 006253          EHALT            ;ERROR IN WRITE, AC1=STATUS
24 04043 000401          JMP .+1
25 04044 006254 E13E:    LOOP
26
27 04045 014152          DSZ SECT         ;DECREMENT THE SECTOR
28 04046 000401          JMP .+1          ;NUMBER
29 04047 020772          LDA #,E13,1
30 04050 024056          LDA 1,KB11
31 04051 122400          SUB 1,#
32 04052 040767          STA #,E13,1
33 04053 101404          INC #,#,SZR      ;SEE IF DONE
34 04054 000733          JMP E13          ;NOT YET
```

```
04055 020143           LDA 0,DTYPE      ;SEE THE DISCUSSION PRECEDING
04056 024207           LDA 1,C137       ;TEST 11.  READ ONCE FROM
04057 103103           ADDL 0,0,SNC     ;SECTOR OF CYL=0, HEAD=0.
04060 024211           LDA 1,C277       ;READ SUCCESSIVE SECTORS
04061 044415           STA 1,E14.2      ;IN ORDER FROM HI TO LOW.
04062 125220           MOVZR 1,1        ;IN EACH CASE THE DATA EQUALS
04063 125220           MOVZR 1,1        ;THE COMPLEMENT OF THE
04064 125220           MOVZR 1,1        ;SECTOR NUMBER
04065 125220           MOVZR 1,1
04066 044152           STA 1,SECT       ;STARTING SECTOR #
04067 006250 E14:      JSR @ISET
04070 006272           DOSEK            ;SEEK !
04071 000000           0                ;CYLINDER 0
04072 006253           EHALT            ;ERROR IN SEEK, AC1=STATUS
04073 000413           JMP E14E         ;SKIP TO END OF TEST
04074 006267           READ             ;READ !
04075 006530 E14.1:    PRGEND           ;DATA BUFFER ADDRESS
04076 000017 E14.2:    17               ;DISK ADDRESS (IT CHANGES)
04077 006253           EHALT            ;ERROR IN READ, AC1=STATUS
04100 000406           JMP E14E         ;SKIP TO END OF TEST
04101 020152           LDA 0,SECT       ;GET SECTOR #
04102 100000           COM 0,0
04103 026772           LDA 1,@E14.1     ;GET A WORD READ
04104 122414           SUB# 1,0,SZR     ;DATA ERROR, SEE ABOVE DESCRIPTION
04105 006253           EHALT            ;AC0=GOOD WORD
04106 006254 E14E:     LOOP             ;AC1=BAD

04107 014152           DSZ SECT         ;DECREMENT TO NEXT SECTOR
04110 000401           JMP .+1
04111 020765           LDA 0,E14.2
04112 024056           LDA 1,KB11
04113 122400           SUB 1,0
04114 040762           STA 0,E14.2
04115 101404           INC 0,0,SZR      ;DONE ?
04116 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.
```

```
;0057 .MAIN
;1
;2 04117 102400          SUB 0,0          ;SEE THE DISCUSSION ABOVE.
;3 04120 042151          STA 0,HEAD       ;WRITE ON CYL-0, SECT-0,
;4 04121 020114          LDA 0,C17        ;ONCE FOR EACH HEAD.
;5 04122 044413          STA 0,E15,1      ;DATA WORDS = HEAD #
;6 04123 006250 E15:     JSR @ISET
;7 04124 006272          DOSEK            ;SEEK !
;8 04125 000000          0                ;CYLINDER 0
;9 04126 006253          EHALT            ;ERROR IN SEEK, AC1=STATUS
;10 04127 000411         JMP E15E         ;SKIP TO END OF TEST
;11 04130 006266         GENDAT           ;GENERATE DATA
;12 04131 005227         HDN              ;ADDRESS OF DATA GENERATOR
;13 04132 006030         PRGEND           ;DATA BUFFER ADDRESS
;14 04133 006270         WRITE            ;WRITE !
;15 04134 006030         PRGEND           ;DATA BUFFER ADDRESS
;16 04135 000017 E15,1:  17               ;DISK ADDRESS (IT CHANGES)
;17 04136 006253         EHALT            ;ERROR IN WRITE, AC1=STATUS
;18 04137 000421         JMP .+1
;19 04140 006254 E15E:   LOOP
;20
;21 04141 010151         ISZ HEAD         ;INCREMENT TO NEXT HEAD
;22 04142 020773         LDA 0,E15,1
;23 04143 024002         LDA 1,K87
;24 04144 123000         ADD 1,0
;25 04145 040770         STA 0,E15,1
;26 04146 024143         LDA 1,DTYPE      ;DONE ?
;27 04147 030454         LDA 2,ADR-1
;28 04150 125112         MOVL# 1,1,SZC
;29 04151 000404         JMP .+4          ;CART DISK
;30 04152 151400         INC 2,2
;31 04153 125282         MOVR 1,1,SZC     ;SKIP IF 2311
;32 04154 151400         INC 2,2          ;2314
;33 04155 025000         LDA 1,0,2        ;GET END DISK ADDR
;34 04156 122404         SUB 1,0,SZR
;35 04157 000744         JMP E15          ;NOT DONE YET
```

```
;0058 .MAIN
;1
;2 04160 102400          SUB 0,0          ;SEE THE DISCUSSION PRECEDING
;3 04161 040151          STA 0,HEAD       ;TEST E15.  READ ONCE FROM
;4 04162 020114          LDA 0,C17        ;EACH HEAD ON CYL-0, SECT-0,
;5 04163 044410          STA 0,E16,2      ;VERIFY THAT DATA=HEAD #
;6 04164 006250 E16:     JSR @ISET
;7 04165 006272          DOSEK            ;SEEK !
;8 04166 000000          0                ;CYLINDER 0
;9 04167 006253          EHALT            ;ERROR IN SEEK, AC1=STATUS
;10 04170 000412         JMP E16E         ;SKIP TO END OF TEST
;11 04171 006267         READ             ;READ !
;12 04172 006030 E16,1:  PRGEND           ;ADDRESS OF DATA BUFFER
;13 04173 000017 E16,2:  17               ;DISK ADDRESS (IT CHANGES)
;14 04174 006253         EHALT            ;ERROR IN READ, AC1=STATUS
;15 04175 000405         JMP E16E         ;SKIP TO END OF TEST
;16 04176 020151         LDA 0,HEAD       ;GET HEAD #
;17 04177 026773         LDA 1,@E16,1     ;GET A WORD READ
;18 04200 122414         SUB# 1,0,SZR     ;DATA ERROR, SEE ABOVE DISCUSSION
;19 04201 006253         EHALT            ;AC0=GOOD WORD
;20 04202 006254 E16E:   LOOP             ;AC1=BAD
;21
;22 04203 010151         ISZ HEAD         ;INCREMENT TO NEXT HEAD
;23 04204 020767         LDA 0,E16,2
;24 04205 024002         LDA 1,K87
;25 04206 123000         ADD 1,0
;26 04207 040764         STA 0,E16,2
;27 04210 024143         LDA 1,DTYPE      ;DONE YET ??
;28 04211 030412         LDA 2,ADR-1
;29 04212 125112         MOVL# 1,1,SZC
;30 04213 000404         JMP .+4          ;CART DISK
;31 04214 151400         INC 2,2
;32 04215 125282         MOVR 1,1,SZC     ;SKIP IF 2311
;33 04216 151400         INC 2,2          ;2314
;34 04217 025000         LDA 1,0,2        ;GET ENDING DISK ADDRESS
;35 04220 122404         SUB 1,0,SZR
;36 04221 000743         JMP E16          ;MORE TO GO
;37 04222 002405         JMP .+5          ;GO TO NEXT TEST
;38
;39 04223 004224         .+1
;40 04224 001017 ADR:    1017             ;CART,  HEAD 2  SECT 0  1 SECT
;41 04225 005017         5217             ;2311,  HEAD 12 SECT 0  1 SECT
;42 04226 012017         12017            ;2314,  HEAD 24 SECT 5  1 SECT
```

```
;0059 .MAIN
#1
02 04227 020143          LDA 0,DTYPE      ;SEE THE DISCUSSION
03 04230 030506          LDA 2,ADR1-1     ;PRECEDING TEST E15.
04 04231 101112          MOVL# 0,0,SZC    ;WRITE ON CYL-0, SECT-0,
05 04232 000404          JMP .+4          ;ONCE FOR EACH HEAD.
06 04233 151400          INC 2,2          ;DATA = HEAD #
07 04234 101200          MOVR 0,0,SZC
08 04235 151400          INC 2,2
09 04236 025000          LDA 1,0,2        ;STARTING DISK ADDRESS
10 04237 044416          STA 1,E17,1      ;SELECT HEADS IN REVERSE ORDER
11 04240 020216          LDA 0,C1774
12 04241 123700          ANDS 1,0
13 04242 040151          STA 0,HEAD       ;CURRENT HEAD #
14 04243 006250 E171     JSR @ISET
15 04244 006272          DOSEK            ;SEEK !
16 04245 000000          0                ;CYLINDER 0
17 04246 006253          EHALT            ;ERROR IN SEEK, AC1=STATUS
18 04247 000411          JMP E17E         ;SKIP TO END OF TEST
19 04250 006266          GENDAT           ;GENERATE DATA
20 04251 005225          MONC             ;ADDRESS OF DATA GENERATOR
21 04252 006530          PRGEND           ;DATA BUFFER ADDRESS
22 04253 006270          WRITE            ;WRITE !
23 04254 006530          PRGEND           ;DATA BUFFER ADDRESS
24 04255 000017 E17,1:   17               ;DISK ADDRESS (IT CHANGES)
25 04256 006253          EHALT            ;ERROR IN WRITE, AC1=STATUS
26 04257 000401          JMP .+1
27 04260 006254 E17E:    LOOP
28
29 04261 014151          DSZ HEAD
30 04262 000401          JMP .+1
31 04263 020772          LDA 0,E17,1      ;DECREMENT HEAD #
32 04264 024062          LDA 1,K87
33 04265 122400          SUB 1,0
34 04266 040767          STA 0,E17,1
35 04267 101103          MOVL 0,0,SNC     ;DONE YET ?
36 04270 000753          JMP E17          ;NO
```

```
;0060 .MAIN
#1
02 04271 020143          LDA 0,DTYPE      ;SEE THE DISCUSSION PRECEDING
03 04272 030444          LDA 2,ADR1-1     ;TEST E15.  READ ONCE FROM
04 04273 101112          MOVL# 0,0,SZC    ;EACH HEAD ON CYL-0, SECT-0.
05 04274 000404          JMP .+4          ;VERIFY THAT DATA=HEAD #
06 04275 151400          INC 2,2
07 04276 101200          MOVR 0,0,SZC     ;SKIP IF 2311
08 04277 151400          INC 2,2          ;2314
09 04300 025000          LDA 1,0,2        ;GET DISK ADDRESS TO BEGIN WITH
10 04301 044413          STA 1,E18,2
11 04302 020216          LDA 0,C1774
12 04303 123700          ANDS 1,0
13 04304 040151          STA 0,HEAD       ;BEGINNING HEAD #
14 04305 006250 E18:     JSR @ISET
15 04306 006272          DOSEK            ;SEEK !
16 04307 000000          0                ;CYLINDER 0
17 04310 006253          EHALT            ;ERROR IN SEEK, AC1=STATUS
18 04311 000413          JMP E18E         ;SKIP TO END OF TEST
19 04312 006267          READ             ;READ !
20 04313 006530 E18,1:   PRGEND           ;ADDRESS OF DATA BUFFER
21 04314 000017 E18,2:   17               ;DISK ADDRESS (IT CHANGES)
22 04315 006253          EHALT            ;ERROR IN READ, AC1=STATUS
23 04316 000406          JMP E18E         ;SKIP TO END OF TEST
24 04317 020151          LDA 0,HEAD       ;GET HEAD #
25 04320 100000          COM 0,0          ;USE THE COMP.
26 04321 025772          LDA 1,@E18,1     ;GET A WORD READ
27 04322 122414          SUB# 1,0,SZR     ;DATA ERROR, SEE ABOVE DISCUSSION
28 04323 000253          EHALT            ;AC0=GOOD WORD
29 04324 006254 E18E:    LOOP             ;AC1=BAD
30
31 04325 014151          DSZ HEAD
32 04326 000401          JMP .+1
33 04327 020755          LDA 0,E18,2      ;DECREMENT HEAD #
34 04330 024062          LDA 1,K87
35 04331 122400          SUB 1,0
36 04332 040762          STA 0,E18,2
37 04333 101103          MOVL 0,0,SNC     ;DONE YET ?
38 04334 000751          JMP E18          ;NO
39 04335 000405          JMP E19          ;YES, GO TO NEXT TEST
40
41 04336 004337          .+1
42 04337 000417 ADR1:    417              ;CART,  HEAD 1  SECT 0  1 SECT
43 04340 004417          4417             ;2311,  HEAD 11 SECT 0  1 SECT
44 04341 011417          11417            ;2314,  HEAD 23 SECT 0  1 SECT
```

```
;0061 .MAIN
;1
;2 04342 006251 E19:    SETP1           ;CAUSE SEEK ERROR BY
;3 04343 006272         DOSEK           ;SEEKING TO CYL 313
;4 04344 000313         313
;5 04345 000423         JMP .+3
;6 04346 000402         JMP .+2         ;AC1=STATUS
;7 04347 006253         EHALT           ;NO ERRORS ON SEEK
;8 04350 006254         LOOP            ;TO CYL 313
;9
;10 04351 030146        LDA 2,UNUM      ;RECALIBRATE THE
;11 04352 021100        LDA 0,TRCL,2    ;UNIT WITH A SEEK ERR
;12 04353 040401        STA 0,.+1
;13 04354 006242        RECLR
;14
;15 04355 006251 E20:    SETP1          ;CAUSE SEEK ERROR BY
;16 04356 006272         DOSEK          ;SEEKING TO CYL 313
;17 04357 000313         313
;18 04360 020057         LDA 0,KB10
;19 04361 123415         ANDZ# 1,0,SNR  ;AC1=STATUS
;20 04362 006253         EHALT          ;NO SEEK ERROR STATUS
;21 04363 006254         LOOP           ;FOLLOWING SEEK TO CYL 313
;22
;23 04364 030146         LDA 2,UNUM     ;RECALIBRATE THE UNIT
;24 04365 021100         LDA 0,TRCL,2   ;WITH A SEEK ERROR
;25 04366 040401         STA 0,.+1
;26 04367 006242         RECLR
;27
;28 04370 006251 E21:    SETP1          ;CHECK FOR ILLEGAL ERRORS
;29 04371 006272         DOSEK          ;ALONG WITH SEEK ERROR
;30 04372 000313         313            ;AC1=STATUS
;31 04373 020172         LDA 0,C36      ;INTENTIONAL SEEK ERROR
;32 04374 123414         ANDZ# 1,0,SZR  ;ILLEGAL STATUS; "END CYL"
;33 04375 006253         EHALT          ;OR "UNSAFE" OR "CHK WD"
;34 04376 006254         LOOP           ;OR "DATA LATE"
;35
;36 04377 030146         LDA 2,UNUM     ;RECALIBRATE THE UNIT
;37 04400 021100         LDA 0,TRCL,2   ;WITH A SEEK ERROR
;38 04401 040401         STA 0,.+1
;39 04402 006242         RECLR
```

```
;0062 .MAIN
;1
;2 04403 006264 E22:     DORW           ;SETUP SECTOR 3 & 4
;3 04404 000000          0              ;CYL 0
;4 04405 005200          THREE          ;DATA TYPE
;5 04406 000077          77             ;DISK ADDRESS
;6
;7 04407 006264          DORW           ;SECT 4
;8 04410 000000          0
;9 04411 005202          FOUR
;10 04412 000117         117
;11
;12 04413 006250         JSR #ISET      ;TEST READ SECT 3-WRITE SECT 4
;13                                      ;SEQUENCE
;14 04414 006267         READ           ;READ A SECTOR
;15 04415 006630         PRGEND         ;MEM BUFFER ADDRESS
;16 04416 000077         77             ;SECTOR THREE
;17 04417 006253         EHALT          ;ERROR, AC1=STATUS
;18 04420 000422         JMP E22E       ;SKIP TO END OF TEST
;19
;20 04421 006270         WRITE          ;WRITE A SECTOR
;21 04422 006630         PRGEND         ;BUFF ADDR (3'S JUST READ)
;22 04423 000117         117            ;SECTOR 4
;23 04424 006253         EHALT          ;ERROR, AC1=STATUS
;24 04425 000415         JMP E22E       ;SKIP TO END OF TEST
;25
;26 04426 006266         GENDAT         ;GENERATE DATA BUFFER
;27 04427 005200         THREE          ;ADDR OF DATA GEN
;28 04430 006630         PRGEND         ;BUFFER ADDR.
;29
;30 04431 006267         READ           ;READ A SECTOR
;31 04432 007230         PRGEND+400     ;BUFF ADDR.
;32 04433 000117         117            ;SECTOR 4
;33 04434 006253         EHALT          ;ERROR, AC1=STATUS
;34 04435 000405         JMP E22E       ;SKIP TO END OF TEST
;35
;36 04436 006271         CHECK          ;COMPARE BUFFERS A/B
;37 04437 006630         PRGEND         ;ADDR OF BUFF A (CORRECT)
;38 04440 007230         PRGEND+400     ;ADDR OF BUFF B
;39 04441 006253         EHALT          ;COMPARE ERROR, AC0=GOOD
;40 04442 000234 E22E:    LOOP          ;AC1=BAD
```

```
;0063 .MAIN
;1
;2 04443 006264 E23:    DORW            ;SETUP SECTOR 3 & 4
;3 04444 000300          0               ;CYL # 0
;4 04445 005200          THREE           ;DATA TYPE
;5 04446 000077          77              ;DISK ADDRESS (SECT 3)
;6
;7 04447 006264          DORW
;8 04450 000300          0
;9 04451 005202          FOUR
;10 04452 000117         117             ;SECTOR 4
;11
;12 04453 006250         JSR @ISET       ;TEST WRITE SECT 3-READ SECT 4
;13                                      ;SEQUENCE.
;14 04454 006266         GENDAT          ;GENERATE DATA
;15 04455 005202         FOUR            ;4'S
;16 04456 006630         PRGEND          ;ADDR OF DATA BUFF
;17
;18 04457 006270         WRITE           ;WRITE A SECTOR
;19 04460 006630         PRGEND          ;ADDR OF DATA BUFF
;20 04461 000077         77              ;SECTOR 3
;21 04462 006253         EHALT           ;ERROR, AC1=STATUS
;22 04463 000417         JMP E23E        ;SKIP TO END OF TEST
;23
;24 04464 006267         READ            ;READ A SECTOR
;25 04465 007230         PRGEND+400      ;MEM BUFF ADDRESS
;26 04466 000117         117             ;SECTOR 4
;27 04467 006253         EHALT           ;ERROR, AC1=STATUS
;28 04470 000412         JMP E23E        ;SKIP TO END OF TEST
;29
;30 04471 006267         READ            ;READ A SECTOR
;31 04472 006630         PRGEND          ;MEM ADDR
;32 04473 000077         77              ;SECTOR 3
;33 04474 006253         EHALT           ;ERROR, AC1=STATUS
;34 04475 000405         JMP E23E        ;SKIP TO END OF TEST
;35
;36 04476 006271         CHECK           ;COMPARE SECTOR 3 & 4
;37 04477 006630         PRGEND          ;SECT 3
;38 04500 007230         PRGEND+400      ;SECT 4
;39 04501 006253         EHALT           ;COMPARE ERROR, ACB=GOOD (SECT 3)
;40 04502 006254 E23E:   LOOP            ;AC1=BAD (SECT 4)
;41 04503 102000         ADC 0,0
;42 04504 040545         STA 0,SSKFLG
;43 04505 000426         JMP E24
;44 04506 102440 RANSK:  SUBO 0,0
;45 04507 040542         STA 0,SSKFLG
;46 04510 006262         INIT
;47 04511 024010         LDA 1,C2        ;SET COUNT FOR
;48 04512 044540         STA 1,SSCNT     ;5000 SEEKS
```

```
;0064 .MAIN
;1 04513 102400 E24:     SUB 0,0         ;WRITE CYL # INTO EACH
;2 04514 040405          STA 0,.CL       ;SECT 0, HEAD 0, OF ALL
;3 04515 040420          STA 0,.SCYL     ;CYLINDERS
;4 04516 024212          LDA 1,C312
;5 04517 044435          STA 1,LCYL
;6 04520 006264          DORW
;7 04521 000000 .CL:     0               ;CYL #
;8 04522 005212          CYLN            ;ADDR OF DATA GEN ROUT.
;9 04523 000017          17              ;DISK ADDRESS
;10
;11 04524 010775         ISZ .CL
;12 04525 020774         LDA 0,.CL
;13 04526 024212         LDA 1,C312
;14 04527 122437         SUBZ# 1,0,SBN
;15 04530 000770         JMP .CL-1
;16 04531 020441         LDA 0,C500
;17 04532 040441         STA 0,CCNT      ; DO 500 SEEKS
;18
;19                      ;AT 5 TIMES EACH BETWEEN RANDOM
;20                      ;CYLINDER NUMBERS
;21 04533 006251 GO:     JSR @I.SETP
;22 04534 006272         DOSEK           ;SEEK !!
;23 04535 000000 .SCYL:  0               ;CYL #
;24 04536 005253         EHALT           ;ERROR, AC1=STATUS
;25 04537 000436         JMP E24R        ;SKIP TO END OF TEST
;26
;27 04540 006267         READ            ;READ DATA IN SECT 0
;28 04541 006630         PRGEND          ;MEM ADDR
;29 04542 000017         17              ;DISK ADDR
;30 04543 006253         EHALT           ;ERROR, AC1=STATUS
;31 04544 000431         JMP E24R        ;SKIP TO END OF TEST
;32
;33 04545 020150         LDA 0,CYL       ;CURRENT CYLINDER #
;34 04546 026153         LDA 1,@BUFF     ;ACTUAL CYL #
;35 04547 122415         SUB# 1,0,SNR    ;(LCYL) = LAST CORRECT
;36 04550 000403         JMP .+3         ;NO ERR
;37 04551 006253         EHALT           ;CYLINDER #
;38 04552 000423         JMP E24R
;39 04553 006272         DOSEK           ;SEEK LAST CYL
;40 04554 000000 LCYL:   0               ;CHANGES !
;41 04555 006253         EHALT
;42 04556 000417         JMP E24R
;43 04557 006267         READ            ;READ SEC 0
;44 04560 006630         PRGEND
;45 04561 000017         17
;46 04562 006253         EHALT
;47 04563 000412         JMP E24R        ;MAYBE ADRS ERR
;48 04564 020150         LDA 0,CYL       ;CURCYL #
;49 04565 026153         LDA 1,@BUFF     ;WHERE WE REALLY ARE
;50 04566 122415         SUB# 1,0,SNR    ;SHD BE SAME
;51 04567 000412         JMP E24E
;52 04570 006253         EHALT
;53 04571 000404         JMP E24R
;54 04572 000764 C500:   500.
;55 04573 000000 CCNT:   0
;56 04574 000002 SKIP:   2               ;SEEK ITERATION COUNT (2 X 2500).
```

```
!0065 .MAIN
;1              ;E24R IS TO RECALIBRATE DRIVE AFTER SEEK ERROR
;2  04575 034146 E24R:   LDA 2,UNUM
;3  04576 021146         LDA 0,TMCL,2
;4  04577 044401         STA 0,.+1
;5  04600 006242         RECLR
;6  04601 006254 E24E:   LOOP
;7  04602 020752         LDA 0,LCYL       ;SAVE CYLINDER JUST DONE
;8  04603 042732         STA 0,.SCYL
;9  04604 006265 GRAN:   JSR @IRAN        ;GET RANDOM CYLINDER #
;10 04605 024212         LDA 1,C312
;11 04606 030120         LDA 2,C377       ;# MUST BE <313
;12 04607 143402         AND 2,0
;13 04610 100433         SUBZ# 0,1,SNC
;14 04611 000775         JMP GRAN         ;TRY AGAIN
;15 04612 042742         STA 0,LCYL
;16 04613 014760         DSZ CCNT
;17 04614 000717         JMP GO
;18 04615 020434         LDA 0,SSKFLG     ;TEST THE SPECIAL SEEK FLAG,
;19 04616 101004         MOV 0,0,SZR      ;IF NON-ZERO, PGM IS NORMAL
;20 04617 000415         JMP E24X         ;DIAGNOSTIC, IF 0 IT IS SEEK
;21 04620 014432         DSZ SSCNT        ;TEST LOOP. COUNT 5000 SEEK 5
;22 04621 000402         JMP .+2          ;
;23 04622 000404         JMP .+4          ;
;24 04623 020747         LDA 0,C500       ;RELOAD COUNT AND LOOP'.
;25 04624 040747         STA 0,CCNT       ;
;26 04625 000765         JMP GO           ;
;27 04626 020746         LDA 0,SKITR      ;RELOAD MULTIPLIER COUNTER'.
;28 04627 042423         STA 0,SSCNT      ;
;29 04630 006260         PCRLF            ;PRINT OUT
;30 04631 006261         MESSAGE          ;A MESSAGE INDICATING
;31 04632 006570         MSG6             ;"PASS"
;32 04633 000770         JMP .-10         ;THEN LOOP AGAIN.
;33 04634 020145 E24X:   LDA 0,TESTU
;34 04635 024047         LDA 1,XB1
;35 04636 010146         ISZ UNUM         ;CHECK TO REPEAT TSTS
;36 04637 123022         ADDZ 1,0,SZC     ;STEP TO NEXT UNIT
;37 04640 000413         JMP E24Y
;38 04641 040145         STA 0,TESTU      ;SAVE UNIT
;39 04642 032146         LDA 2,UNUM
;40 04643 025052         LDA 1,UNTBIT,2   ;GET UNIT BIT
;41 04644 020144         LDA 0,NOSKS      ;AND EXISTING BITS
;42 04645 107425         AND 0,1,SNR      ;SKP=DRIVE EXISTS
;43 04646 000766         JMP E24X         ;CHK ON NEXT DRV
;44 04647 000401         JMP 0,.+1
;45 04650 000200         DI               ;REPEAT TESTS NEXT DRV'
;46 04651 000000 SSKFLG: 0
;47 04652 000000 SSCNT:  0
;48 04653 152420 E24Y:   SUB 2,2          ;RESET TO LOWEST
;49 04654 021052         LDA 0,UNTBIT,2   ;EXISTING DRV
;50 04655 025046         LDA 1,.TU,2
;51 04656 034144         LDA 3,NOSKS      ;EXIST BITS
;52 04657 163404         AND 3,0,SZR      ;SKP=NO DRV
;53 04660 000403         JMP .+3
;54 04661 151400         INC 2,2
;55 04662 000772         JMP E24Y+1       ;TRY NEXT
;56 04663 044145         STA 1,TESTU
;57 04664 050146         STA 2,UNUM
```

```
!0066 .MAIN
;1               ;THE FOLLOWING TESTS CHECK FOR THE CORRECT OPER.
;2               ;OF THE FOLLOWING DISK FUNCTIONS:
;3               ;     1.THAT A WRITE INTO END OF CYLINDER
;4               ;        A. WILL CAUSE AN END OF CYLINDER ERROR
;5               ;        B. DOES NOT TURN THE WRITE FUNCTION OFF
;6               ;           UNTIL THE CHECKSUM HAS BEEN COMPLETELY WRITTEN.
;7               ;           (SEE INC. SC. DELAY)
;8               ;        C.  DOES TURN THE WRITERS OFF COMPLETLY SO THAT THE
;9               ;           NEXT HEAD SELECTION DOES NOT TURN THE WRITE
;10              ;           CURRENT ON WHERE IT IS NOT WANTED.
;11              ;           (SEE DIRECT CLEAR INPUT TO WREN.)
;12 04665 102420          SUBZ 0,0
;13 04666 040511          STA 0,F25D
;14 04667 101100          MOVL 0,0
;15 04670 040506          STA 0,F25M       ;DRIVE MASK
;16 04671 006252 E25:     SETUP
;17 04672 020143          LDA 0,DTYPE
;18 04673 030505          LDA 2,E25T       ;HEAD ADRS TBLE
;19 04674 101112          MOVL# 0,0,SZC
;20 04675 000404          JMP .+4          ;CARTRIDGE
;21 04676 151400          INC 2,2
;22 04677 101002          MOVR 0,0,SZC     ;SKP IS 2311
;23 04700 151400          INC 2,2          ;2314 IS THE TYPE
;24 04701 025500          LDA 1,0,2        ;GET LAST HEAD #
;25 04702 044502          STA 1,E25H       ;AND SAVE IT
;26 04703 030473          LDA 2,E25H
;27 04704 020144          LDA 0,NOSKS
;28 04705 113405          AND 0,2,SNR      ;THIS DISK EXIST?
;29 04706 000457          JMP E25L+1       ;NO TRY NEXT
;30 04707 020470          LDA 0,E25D
;31 04710 053233          DOCC 0,DSKP      ;SELECT CURRENT DISK
;32 04711 024063          LDA 1,KB6
;33 04712 065333          DOAP 1,DSKP      ;STRT SEEK 0
;34 04713 030161          LDA 2,C4         ;WAIT 400 MS
;35 04714 006240          WAIT
;36 04715 020462          LDA 0,E25D
;37 04716 024466          LDA 1,E25H
;38 04717 107000          ADD 0,1
;39 04720 067233          DOCC 1,DSKP      ;DRV LAST NO SEC IS .2 SECTORS
;40               ;UNLESS 2311 SECTOR 5,2 SECTORS
;41 04721 020050          LDA 0,KB7        ;"WRITE"
;42 04722 061033          DOA 0,DSKP
;43 04723 062133          DOBS 0,DSKP      ;STRTWRITE 2 SECTORS
;44 04724 030053          LDA 2,KB14       ;*200 MS
;45 04725 006240          WAIT
;46 04726 060433          DIA 0,DSKP
;47 04727 105223          MOVZR 0,1,SNC    ;ERR SHD=1
;48 04730 000404          JMP E25E1
;49 04731 024456          LDA 1,KB11       ;END BIT SHD =1
;50 04732 123414          ANDM 1,0,SZR     ;SKP IF IT DSNIT
;51 04733 000403          JMP .+3
;52 04734 006253 E25E1:   EHALT            ;END OF CYLINDER NOT FOUND
;53 04735 000427          JMP E25L         ;DO LOOP
;54               ;WRITE INTO END OF SECTOR AT LEAST APPEARS CORRECT
;55               ;SEE IF THE DATA CAN BE READ CORRECTLY
```

```
10067 .MAIN
01                ;READ SEC 13 OF LAST HEAD FOR NO CHKWRD ERR
02                ;OR SECTOR 5 OF LAST HEAD IF 2311
03 04736 020153           LDA 0,BUFF
04 04737 020445           LDA 1,E25M
05 04740 030437           LDA 2,E250
06 04741 147000           ADD 2,1
07 04742 152400           SUB 2,2
08 04743 062233           DOBC 0,DSKP
09 04744 062033           DOC 1,DSKP
10 04745 071133           DOAS 2,DSKP       ;STRT 2 CYL READ
11 04746 030053           LDA 2,KB14
12 04747 006240           WAIT
13 04750 060433           DIA 0,DSKP        ;GET STATUS
14 04751 101113           MOVL# 0,0,SNC     ;READ DONE?
15 04752 000411           JMP E25E2         ;NO
16 04753 105203           MOVR 0,1,SNC      ;ERR SHD =1
17 04754 000407           JMP E25E2         ;EOC SHD =1
18 04755 030055           LDA 2,KB12        ;END BIT
19 04756 147415           AND# 2,1,SNR      ;SHD =1 FOR EOC
20 04757 000404           JMP E25E2         ;BUT DOESN'T
21 04760 125200           MOVR 1,1
22 04761 125203           MOVR 1,1,SNC      ;SKP HERE IS ERR
23 04762 000402           JMP .+2           ;OK NO CHKWORD ERR
24 04763 006253 E25E2:    EMALT             ;SEE AC0 FOR DSK STAT
25                ;EITTHER A, NO READ DONE  B, NO EOC C, A CHKWORD ERR
26                ;                 BIT 0=0 BIT 11=0 OR BIT 13=1
27 04764 006254 E25L:     LOOP
28 04765 020412           LDA 0,E25D
29 04766 024047           LDA 1,KB1
30 04767 123822           ADDZ 1,0,SZC      ;+1 TILL DRV 3 DONE
31 04770 000415           JMP NDMES
32 04771 040400           STA 0,E25D
33 04772 020404           LDA 0,E25M
34 04773 101120           MOVZL 0,0
35 04774 040402           STA 0,E25M
36 04775 000674           JMP E25
37 04776 000000 E25M: 0
38 04777 000000 E25D: 0
39 05000 005001 E25T: .+1
40 05001 000676           676      ;CARTRIDGE
41 05002 004536           4536     ;2311 LAST HEAD
42 05003 011676           11676    ;2314 HD 23
43 05004 000000 E25M: 0
```

```
10068 .MAIN
01
02 05005 000260 NDMES:  PCRLF             ;END TEST
03 05006 000261         MESSAGE
04 05007 000570         MSG6              ;"PASS"
05 05010 000401         JMP .+1
06 05011 000003         A0
07
08
09
```

```
!0069 .MAIN
01
02                      ;RANDOM NUMBER GENERATOR
03
04 05012 054431 RAN:    STA 3,.UD03     ;GENERATE A RANDOM
05 05013 052427         STA 2,.UD02
06 05014 044425         STA 1,.UD01
07 05015 020135         LDA 0,RANDOM    ;NUMBER IN AC0
08 05016 004410         JSR .UD50
09 05017 034426         LDA 3,.UD20
10 05020 163000         ADD 3,0
11 05021 042135         STA 0,RANDOM    ;STORE NEW VALUE.
12 05022 111100         MOVL 0,2
13 05023 030417         LDA 2,.UD02
14 05024 024415         LDA 1,.UD01
15 05025 002416         JMP @.UD03
16
17 05026 024420 .UD50:  LDA 1,.UD21     ;RANDOM CONTINUED
18 05027 044415         STA 1,.UD10
19 05030 105120         MOVZL 0,1
20 05031 125120         MOVZL 1,1
21 05032 014412         DSZ ,UD10
22 05033 000770         JMP .-2
23 05034 107000         ADD 0,1
24 05035 125120         MOVZL 1,1
25 05036 125120         MOVZL 1,1
26 05037 123000         ADD 1,0
27 05040 001400         JMP 0,3
28 05041 000000 .UD01:  0
29 05042 000000 .UD02:  0
30 05043 000000 .UD03:  0
31 05044 000000 .UD10:  0
32 05045 033031 .UD20:  33031
33 05046 000010 .UD21:  10
```

```
!0070 .MAIN
01
02                      ;CHECK DATA SUBROUTINE
03                      ; CALL  CHECK
04                      ;       ADDRESS OF DATA BUFFER 1
05                      ;       ADDRESS OF DATA BUFFER 2
06                      ;       ERROR RETURN, (AC1)=BAD (AC0)=GOOD
07                      ;       NORMAL RETURN
08
09 05047 054460 .CHECK: STA 3,GENRET
10 05050 030221         LDA 2,M400
11 05051 050417         STA 2,CTR
12 05052 031400         LDA 2,0,3
13 05053 035401         LDA 3,1,3
14 05054 010453         ISZ GENRET
15 05055 010452         ISZ GENRET
16 05056 021000 .CHE1:  LDA 0,0,2
17 05057 025400         LDA 1,0,3
18 05060 106414         SUBN 0,1,SZR
19 05061 002446         JMP @GENRET     ;ERROR
20 05062 151400         INC 2,2
21 05063 175400         INC 3,3
22 05064 010404         ISZ CTR
23 05065 000771         JMP .CHE1       ;CHECK MORE
24 05066 010441         ISZ GENRET
25 05067 002440         JMP @GENRET     ;NORMAL RETURN
26 05070 000000 CTR:    0
27
28                      ;GENERATE ONE SECTOR OF DATA
29                      ; CALL  GENDAT
30                      ;       ADDRESS OF DATA GEN ROUTINE
31                      ;       DATA BUFFER ADDRESS
32                      ;       RETURN
33
34 05071 054436 .GEN:   STA 3,GENRET
35 05072 024221         LDA 1,M400
36 05073 031401         LDA 2,1,3
37 05074 034433 .GEN1:  LDA 3,GENRET
38 05075 007400         JSR @0,3        ;GET A DATA WORD
39 05076 041000         STA 0,0,2
40 05077 151400         INC 2,2
41 05100 125404         INC 1,1,SZR
42 05101 000773         JMP .GEN1       ;DO MORE
43 05102 034425         LDA 3,GENRET    ;DONE
44 05103 001402         JMP 2,3
```

```
1#071 .MAIN
#1
#2                      ;WRITE SUBROUTINE
#3                      ; CALL  WRITF
#4                      ;       DATA BUFFER ADDRESS
#5                      ;       DISK ADDRESS
#6                      ;       ERROR RETURN, (AC1) = STATUS
#7                      ;       JMP TO END OF TEST
#8                      ;       NORMAL RETURN
#9
10                      ;ERROR RETURN IF
11                      ;       TIMEOUT (100MS)
12                      ;       DATA LATE
13                      ;       ADDRESS ERROR/UNSAFE
14                      ;       END CYLINDER
15                      ;       SEEK ERROR
16                      ;       ANY "SEEKING"
17                      ;       ANY "SEEK DONE"
18                      ;       NO "R/W DONE"
19
20 05104 054423 .WRITE: STA 3,GENRET
21 05105 021400         LDA 0,0,3
22 05106 062033         DOB 0,DSKP      ;CA
23 05107 020145         LDA 0,TESTU
24 05110 025401         LDA 1,1,3
25 05111 123000         ADD 1,0
26 05112 063233         DOCC 0,DSKP     ;UNIT # & DISK ADDRESS
27 05113 020121         LDA 0,C777      ;BIT 7 AND ALL CYL BITS=1
28 05114 061133         DOAS 0,DSKP     ;WRITE !
29 05115 010412         ISZ GENRET
30 05116 010411         ISZ GENRET
31 05117 006246         ITRWT           ;WAIT 100MS FOR INTERRUPT
32 05120 002427         JMP @GENRET     ;TIMEOUT
33 05121 020407 .WR1:   LDA 0,.WM       ;ERROR BIT MASK
34 05122 107415         ANDM 0,1,SNR
35 05123 125113         MOVL# 1,1,SNC
36 05124 002403         JMP @GENRET     ;ERROR STATUS
37 05125 034402         LDA 3,GENRET
38 05126 001402         JMP 2,3
39
40 05127 000000 GENRET: 0
41 05130 077677 .WM:    77677
```

```
1#072 .MAIN
#1
#2                      ;READ SUBROUTINE
#3                      ; CALL  READ
#4                      ;       DATA BUFFER ADDRESS
#5                      ;       DISK ADDRESS
#6                      ;       ERROR RETURN, (AC1)=STATUS
#7                      ;       JMP TO END TEST
#8                      ;       NORMAL RETURN
#9
10                      ;ERROR RETURN IF
11                      ;       TIMEOUT (100MS)
12                      ;       DATA LATE
13                      ;       CHECK WORD ERROR
14                      ;       ADDRESS ERROR/UNSAFE
15                      ;       END CYLINDER
16                      ;       SEEK ERROR
17                      ;       ANY "SEEKING"
18                      ;       ANY "SEEK DONE"
19                      ;       NO "R/W DONE"
20
21 05131 054776 .READ:  STA 3,GENRET
22 05132 021400         LDA 0,0,3
23 05133 062033         DOB 0,DSKP      ;CA
24 05134 020145         LDA 0,TESTU
25 05135 025401         LDA 1,1,3
26 05136 123000         ADD 1,0
27 05137 063233         DOCC 0,DSKP     ;UNIT # & DISK ADDRESS
28 05140 102400         SUB 0,0
29 05141 061133         DOAS 0,DSKP     ;READ !
30 05142 010765         ISZ GENRET
31 05143 010764         ISZ GENRET
32 05144 006246         ITRWT           ;WAIT 100 MS FOR INTERRUPT
33 05145 002762         JMP @GENRET     ;TIMEOUT
34 05146 000753         JMP .WR1        ;CHECK STATUS
```

```
[0073 .MAIN
01
02                      ;SEEK SUBROUTINE
03                      ; CALL  DOSEK
04                      ;       CYL #
05                      ;       ERROR RETURN (AC1)=STATUS
06                      ;       JMP TO END TEST
07                      ;       NORMAL RETURN
08
09                      ;ERROR RETURN IF
10                      ;       TIMEOUT (500MS)
11                      ;       DATA LATE
12                      ;       CHECK WORD ERROR
13                      ;       ADDRESS ERROR/UNSAFE
14                      ;       END CYLINDER
15                      ;       SEEK ERROR
16                      ;       ANY "SEEKING"
17                      ;       NO "SEEK DONE"
18                      ;       "R/W DONE"
19
20 05147 054760 .DOSEK: STA 3,GENRET
21 05150 010757          ISZ GENRET
22 05151 020145          LDA 0,TESTU
23 05152 063233          DOCC 0,DSKP     ;SELECT UNIT
24 05153 021400          LDA 0,0,3
25 05154 024120          LDA 1,C377
26 05155 123400          AND 1,0
27 05156 040150          STA 0,CYL       ;SAVE CYL #
28 05157 024063          LDA 1,K66
29 05160 123000          ADD 1,0         ;CYL # + SEEK
30 05161 063333          DOAP 0,DSKP
31 05162 030162          LDA 2,C5
32 05163 000624          WAIT            ;WAIT 500MS (OR UNTIL "DONE")
33 05164 022410          LDA 0,.DM
34 05165 123414          ANDZ 1,0,SZR
35 05166 002741          JMP @GENRET     ;ERROR STATUS
36 05167 022225          LDA 0,C74K
37 05170 123415          ANDZ 1,0,SNR
38 05171 002736          JMP @GENRET     ;NO SEEK DONE
39 05172 034735          LDA 3,GENRET
40 05173 001402          JMP 2,3
41
42 05174 103677 .DM:    103677
```

```
[0074 .MAIN
01
02 05175 102001 ONES:   AUC 0,0,SKP
03 05176 102400 ZEROS:  SUB 0,0
04 05177 001400         JMP 0,3
05
06 05200 020112 THREE:  LDA 0,C3
07 05201 001400         JMP 0,3
08
09 05202 020101 FOUR:   LDA 0,C4
10 05203 001400         JMP 0,3
11
12 05204 020133 ALT1:   LDA 0,C2525
13 05205 001400         JMP 0,3
14
15 05206 020134 ALT0:   LDA 0,C5252
16 05207 001400         JMP 0,3
17
18 05210 020150 CYLNC:  LDA 0,CYL
19 05211 100001         COM 0,0,SKP
20 05212 020150 CYLN:   LDA 0,CYL
21 05213 001400         JMP 0,3
22
23 05214 054407 NUMSEQ: STA 3,NSRET
24 05215 010407         ISZ NS1
25 05216 000401         JMP .+1
26 05217 020405         LDA 0,NS1
27 05220 034120         LDA 3,C377
28 05221 103400         AND 3,0
29 05222 002401         JMP @NSRET
30 05223 000000 NSRET:  0
31 05224 000000 NS1:    0
32
33 05225 020151 HDNC:   LDA 0,HEAD
34 05226 100001         COM 0,0,SKP
35 05227 020151 HDN:    LDA 0,HEAD
36 05230 001400         JMP 0,3
37
38 05231 020152 SETNC:  LDA 0,SECT
39 05232 100001         COM 0,0,SKP
40 05233 020152 SECTN:  LDA 0,SECT
41 05234 001400         JMP 0,3
```

```
10075 .MAIN
01
02                              ;PROGRAM INITILAIZATION
03 05235 000704                 PTIME   ;CALIBRATION ROUTINE
04 05236 054522 .INI:   STA 3,.INRET
05 05237 022140                 LDA 0,TIME
06 05240 101004                 MOV 0,0,SZR     ;SKIP=NOT CALIB.
07 05241 000400                 JMP .+6
08 05242 000773                 JSR @.INI-1
09 05243 124400                 NEG 1,1
10 05244 044141                 STA 1,TIME1
11 05245 152400                 NEG 2,2
12 05246 054140                 STA 2,TIME
13 05247 006260                 PCRLF
14 05250 006261                 MESSAGE
15 05251 006476                 MSG1            ;DISK PACK TYPE
16 05252 006260                 PCRLF
17 05253 006281                 MESSAGE
18 05254 006511                 MSG2            ;TYPE 0 FOR CART
19 05255 006260                 PCRLF           ;     1 FOR 2311
20 05256 006261                 MESSAGE         ;     2 FOR 2314
21 05257 006527                 MSG3
22
23 05260 006260 .INI1:  PCRLF
24 05261 060210                 NIOC TTI        ;WAIT FOR TTI INPUT
25 05262 060211                 NIOC TTO
26 05263 063610                 SKPDN TTI
27 05264 000777                 JMP .-1
28 05265 060410                 DIA 0,TTI       ;READ CHAR
29 05266 061111                 DOAS 0,TTO      ;ECHO IT
30 05267 063611                 SKPDN TTO
31 05270 000777                 JMP .-1
32 05271 024117                 LDA 1,C177      ;7 BIT MASK
33 05272 107400                 AND 0,1
34 05273 030203                 LDA 2,C60
35 05274 034204                 LDA 3,C63
36 05275 132437                 SUBZ# 1,2,SBN   ;CHAR MUST 0,1, OR 2
37 05276 166432                 SUBZ# 3,1,SZC
38 05277 000452                 JMP QUEST       ;ILLEGAL CHAR TYPED
39 05300 024112                 LDA 1,C3
40 05301 123400                 AND 1,0
41 05302 101224                 MOVZR 0,0,SZR
42 05303 000404                 JMP .+4         ;=2, 2314, BIT 15 IS SET, DONE
43 05304 101205                 MOVR 0,0,SNR
44 05305 101241                 MOVOR 0,0,SKP   ;=0, CART, MUST SET BIT 0
45 05306 101222                 MOVZR 0,0       ;=1, 2311, MUST SET BIT 1
46 05307 040143                 STA 0,DTYPE
```

```
10075 .MAIN
01
02 05310 102400                 SUB 0,0
03 05311 040144                 STA 0,NDSKS
04 05312 006260 .INI2:  PCRLF
05 05313 006251                 MESSAGE
06 05314 006550                 MSG5            ;TYPE UNIT NUMBERS TO TEST
07 05315 006260                 PCRLF
08
09 05316 060210 .INI3:  NIOC TTI
10 05317 060211                 NIOC TTO
11 05320 063610                 SKPDN TTI
12 05321 000777                 JMP .-1
13 05322 060410                 DIA 0,TTI       ;READ CHAR TYPED
14 05323 061111                 DOAS 0,TTO      ;ECHO THE CHAR
15 05324 063611                 SKPDN TTO
16 05325 000777                 JMP .-1
17 05326 024117                 LDA 1,C177
18 05327 107400                 AND 0,1         ;7 BIT ASCII
19 05330 030156                 LDA 2,C15
20 05331 132415                 SUB# 1,2,SNR
21 05332 000422                 JMP CR          ;CR TYPED
22 05333 030223                 LDA 2,C60
23 05334 034204                 LDA 3,C63
24 05335 166437                 SUBZ# 3,1,SBN   ;# MUST BE 0,1,2, OR 3
25 05336 132436                 SUBZ# 1,2,SEZ
26 05337 000426                 JMP OST1        ;ILLEGAL CHAR TYPED
27 05340 030112                 LDA 2,C3
28 05341 113400                 AND 0,2
29 05342 021052                 LDA 0,UNTBIT,2
30 05343 025046                 LDA 1,.TU,2
31 05344 044145                 STA 1,TESTU
32 05345 050146                 STA 2,UNUM
33 05346 110000                 COM 0,2
34 05347 024144                 LDA 1,NDSKS
35 05350 147400                 AND 2,1         ;(NDSKS) = BIT 15 - UNIT 0
36 05351 107000                 ADD 0,1         ;          BIT 14 - UNIT 1
37 05352 044144                 STA 1,NDSKS     ;          BIT 13 - UNIT 2
38 05353 000743                 JMP .INI3       ;          BIT 12 - UNIT 3
39
40 05354 020144 CR:     LDA 0,NDSKS     ;IF NO UNIT NUMBERS TYPED
41 05355 101005                 MOV 0,0,SNR     ;IT IS AN ERROR
42 05356 000407                 JMP OST1
43 05357 002401                 JMP @.INRET
44 05360 000000 .INRET: 0
45
46 05361 006260 QUEST:  PCRLF
47 05362 006261                 MESSAGE
48 05363 006545                 MSG4
49 05364 000674                 JMP .INI1
50
51 05365 006260 OST1:   PCRLF
52 05366 006261                 MESSAGE
53 05367 006545                 MSG4
54 05370 000722                 JMP .INI2
```

```
10077 .MAIN
01
02                      ;SET ALL I/O ADDRESSES
03
04 05371 102401 .S33:   SUB 0,0,SKP     ;SET TO 33
05 05372 000057 .S73:   LDA 0,KB10      ;SET TO 73
06 05373 024171         LDA 1,C33
07 05374 123000         ADD 1,0
08 05375 044147         STA 0,CDSK
09 05376 030422         LDA 2,FIRST
10 05377 021000 .SN0:   LDA 0,0,2
11 05400 024154         LDA 1,MSK1
12 05401 107400         AND 0,1
13 05402 034155         LDA 3,DPIO
14 05403 166424         SUB 3,1,SZR
15 05404 000406         JMP .SN1        ;GO ON
16 05405 024156         LDA 1,MSK2      ;DISK PACK IO INST.
17 05406 034147         LDA 3,CDSK
18 05407 123400         AND 1,0
19 05410 163000         ADD 3,0
20 05411 041000         STA 0,0,2
21 05412 151400 .SN1:   INC 2,2
22 05413 024406         LDA 1,LAST
23 05414 132414         SUB# 1,2,SZR
24 05415 000762         JMP .SN0
25 05416 000401         JMP 0,.+1
26 05417 000502         START
27
28 05420 000502 FIRST:  START
29 05421 005615 LAST:   .SET
30
31                      ;DELAY SUBROUTINE
32
33 05422 054406 .STL:   STA 3,.STLRET
34 05423 034053         LDA 3,KB14
35 05424 054000         STA 3,0
36 05425 014000         DSZ 0
37 05426 000777         JMP .-1
38 05427 002401         JMP 0,STLRET
39 05430 000000 .STLRET:0
```

```
10078 .MAIN
01
02                      ;"SET A SEEKING FLOP" SUBROUTINE
03                      ;FOR THE UNITS INDICATED BY (AC2).
04                      ;       BIT 15 = UNIT 0
05                      ;       BIT 14 = UNIT 1
06                      ;       BIT 13 = UNIT 2
07                      ;       BIT 12 = UNIT 3
08
09 05431 020051 .SSEK:  LDA 0,C140K
10 05432 151005         MOV 2,2,SNR
11 05433 001400         JMP 0,3         ;DONE
12 05434 024047         LDA 1,KB1
13 05435 123000         ADD 1,0
14 05436 151223         MOVZR 2,2,SNC
15 05437 000773         JMP .SSEK+1     ;TRY ANOTHER UNIT
16 05440 063033         DOC 0,DSKP      ;RECAL THIS UNIT
17 05441 024063         LDA 1,C1800
18 05442 065333         DOAP 1,DSKP     ;SEEK !!
19 05443 000767         JMP .SSEK+1
20
21                      ;WAIT ROUTINE FOR "DONE"
22                      ;RUN TIMER
23                      ;(AC2) = MAX RUN TIME. (100MS/COUNT)
24
25 05444 050157 .WAIT:  STA 2,ITRCNT    ;# 100MS ITERATIONS
26 05445 040137         STA 0,TEMP      ;SAV AC0
27 05446 030223         LDA 2,C174K
28 05447 020140 .WTL.:  LDA 0,TIME
29 05450 060000         NIO 0
30 05451 064433         DIA 1,DSKP      ;READ STATUS
31 05452 133414         ANDM 1,2,SZR
32 05453 000405         JMP WTD         ;SOMEONE IS DONE
33 05454 101404         INC 0,0,SZR     ;100 MS CHK
34 05455 000773         JMP .WTL.+1
35 05456 014157         DSZ ITRCNT      ;DONE 100 MS
36 05457 000770         JMP .WTL.       ;DO SOME MORE
37 05460 064433 WTD:    DIA 1,DSKP
38 05461 020137         LDA 0,TEMP
39 05462 001400         JMP 0,3
```

```
10079 .MAIN
01
02
03                      ;RECALIBRATE SUBROUTINE
04
05 05463 102400 .RCL0:  SUB 0,0          ;UNIT 0 ENTRY
06 05464 000405         JMP .RCL3+1
07 05465 020047 .RCL1:  LDA 0,KB1        ;UNIT 1 ENTRY
08 05466 000403         JMP .RCL3+1
09 05467 102621 .RCL2:  SUBZR 0,0,SKP    ;UNIT 2 ENTRY
10 05470 020051 .RCL3:  LDA 0,C140K      ;UNIT 3 ENTRY
11 05471 063233         DOCC 0,DSKP      ;SELECT UNIT
12 05472 022222         LDA 0,C140B
13 05473 063133         DOAP 0,DSKP      ;RECALIBRATE
14 05474 054404         STA 3,RCLRET
15 05475 03W166         LDA 2,C15
16 05476 006240         WAIT             ;WAIT 1.3SEC (OR UNTIL DONE)
17 05477 002401         JMP @RCLRET
18 05500 000000 RCLRET: 0
```

```
10000 .MAIN
01
02                      ;SEEK WRITE/READ/CHECK SUBROUTINE
03                      ;CALL   DORW
04                      ;       CYL#
05                      ;       ADDRESS OF DATA GENERATOR
06                      ;       DISK ADDRESS
07                      ;       RETURN
08
09 05501 054441 .DORW:  STA 3,DRWRET
10 05502 021400         LDA 0,0,3        ;XFER CYL #
11 05503 040410         STA 0,D
12 05504 021401         LDA 0,1,3        ;XFER ADDR DAT GEN
13 05505 040412         STA 0,.D01
14 05506 021402         LDA 0,2,3        ;XFER DISK ADDR
15 05507 040414         STA 0,.D02
16 05510 040420         STA 0,.D03
17
18 05511 006250         JSR @ISET
19 05512 006272         DOSEK            ;SEEK !!
20 05513 000000 D:      0                ;CYL #
21 05514 006253         EHALT            ;ERROR, AC1=STATUS
22 05515 000422         JMP E,DO         ;SKIP TO END OF TEST
23
24 05516 006266         GENDAT           ;GENERATE DATA
25 05517 005175 .D01:   ONES             ;ADDRESS OF DATA GEN
26 05520 006630         PRGEND           ;MEM ADDR
27
28 05521 006270         WRITE            ;WRITE !!
29 05522 006630         PRGEND           ;MEM ADDR
30 05523 000017 .D02:   17               ;DISK ADDRESS
31 05524 006253         EHALT            ;ERROR, AC1=STATUS
32 05525 000412         JMP E,DO         ;SKIP TO END OF TEST
33
34 05526 006267         READ             ;READ !!
35 05527 007230         PRGEND+400       ;MEM ADDR
36 05530 000017 .D03:   17               ;DISK ADDRESS
37 05531 006253         EHALT            ;ERROR, AC1=STATUS
38 05532 000405         JMP E,DO         ;SKIP TO END OF TEST
39
40 05533 006271         CHECK            ;COMPARE BUFFER A/B
41 05534 006630         PRGEND           ; A
42 05535 007230         PRGEND+400       ; B
43 05536 006253         EHALT            ;COMPARE ERROR, AC0=GOOD
44 05537 006254 E,DO:   LOOP             ;AC1=BAD
45
46 05540 034402         LDA 3,DRWRET
47 05541 001403         JMP 3,3
48
49 05542 000000 DRWRET: 0
```

```
0001 .MAIN
01
02                      ;WAIT FOR INTERRUPT TIMEOUT AFTER 100 MS
03                      ;RETURN+1  IF TIMEOUT
04                      ;RETURN+2  IF INTERRUPT
05
06 05543 010277 .IWT:   ISZ SW
07 05544 000415         JMP I2
08 05545 102400 I1:     SUB 0,0
09 05546 040277         STA 0,SW
10 05547 020141         LDA 0,TIME1
11 05550 060177         INTEN   ;ENABLE INTERRUPT
12 05551 101000         MOV 0,0
13 05552 101425         INC 0,0,SNR     ;COUNT 100 MS
14 05553 000403         JMP .+3
15 05554 063400         SKPBN 0         ;TYPE 1 LOOP
16 05555 000774         JMP .-4
17 05556 060277         INTDS           ;NO INTERRUPT
18 05557 064433         DIA 1,DSKP
19 05560 001400         JMP 0,3         ;ERROR RETURN
20
21 05561 020217 I2:     LDA 0,IOMSK
22 05562 062077         MSKO 0
23 05563 000762         JMP I1
24 05564 064433 IRET:   DIA 1,DSKP
25 05565 001401         JMP 1,3         ;NORM INTERRUPT RETURN
26
27                      ;CHOOSE AN ACTIVE DISK UNIT
28                      ;RETURN WITH UNIT # IN AC2
29                      ;       UNIT 0 = 1
30                      ;       UNIT 1 = 2
31                      ;       UNIT 2 = 4
32                      ;       UNIT 3 = 10
33
34 05566 152521 .ADSK:  SUBZL 2,2,SKP
35 05567 151120         MOVZL 2,2
36 05570 020144         LDA 0,NDSKS
37 05571 143405         AND 2,0,SNR
38 05572 000775         JMP .-3
39 05573 001400         JMP 0,3
40
41                      ;SEEK SUBROUTINE
42                      ; CALL  SEEK            ;AC2 = UNIT
43                      ;       N               ;CYL #
44                      ;       RETURN          ;AC1 = STATUS
45
46 05574 054420 .SK:    STA 3,SKRET
47 05575 126400         SUB 1,1
48 05576 022047         LDA 0,KB1
49 05577 151222         MOVZR 2,2,SZC
50 05600 000403         JMP .SK1
51 05601 107000         ADD 0,1
52 05602 000775         JMP .-3
53
54 05603 067033 .SK1:   DOC 1,DSKP      ;SELECT UNIT
55 05604 021400         LDA 0,0,3
56 05605 024063         LDA 1,KB6
57 05606 123000         ADD 1,0         ;SEEK + CYL #
58 05607 061333         DOAP 0,DSKP
59 05610 030162         LDA 2,C5
60 05611 000240         WAIT            ;WAIT 500MS (OR UNTIL DONE)
```

```
0002 .MAIN
01 05612 010402         ISZ SKRET       ;AC1=STATUS
02 05613 002401         JMP @SKRET
03 05614 000000 SKRET:  0
```

```
10083 .MAIN
01
02 05615 054201 .SET:   STA 3,LOOPR      ;ITERATE ONCE
03 05616 176520         SUBZL 3,3
04 05617 000406         JMP .SETUP+2
05
06 05620 054201 .SETP1: STA 3,LOOPR      ;ITERATE 5 TIMES
07 05621 034102         LDA 3,C5
08 05622 000403         JMP .SETUP+2
09
10 05623 054201 .SETUP: STA     3,LOOPR ;ADDRESS OF TOP OF LOOP
11 05624 034415         LDA 3,ITR       ;THIS ROUTINE INITIALIZES
12 05625 062677         IORST
13 05626 040422         STA 0,SAV0
14 05627 044420         STA 1,SAV1
15 05630 050416         STA 2,SAV2
16 05631 175400         INC 3,3
17 05632 054410         STA 3,ITRCT     ;EACH TEST
18 05633 034217         LDA 3,IOMSK
19 05634 076677         MSKO 3
20 05635 176400         SUB 3,3
21 05636 054405         STA 3,ESWIT
22 05637 054405         STA 3,ERRCT
23 05640 000435         JMP .LOP1+1
24
25 05641 000144 ITR:    144
26 05642 000000 ITRCT:  0
27 05643 000000 ESWIT:  0
28 05644 000000 ERRCT:  0
29 05645 000000 .RTRN:  0
30 05646 000000 SAV2:   0
31 05647 000000 SAV1:   0
32 05650 000000 SAV0:   0
33 05651 000000 DSWT:   0
```

```
10084 .MAIN
01
02 05652 054773 .LUPD:  STA 3,.RTRN      ;INTRODUCE A SHORT
03 05653 176401         SUB 3,3,SKP     ;AC3=0 IS WAIT SW.
04
05 05654 054771 .LOOP:  STA 3,.RTRN      ;END OF TEST ITERATION
06 05655 050771         STA 2,SAV2
07 05656 044771         STA 1,SAV1
08 05657 040771         STA 0,SAV0
09 05660 175004         MOV 3,3,SZR     ;INSERT WAIT?
10 05661 000413         JMP .LOP1       ;NO
11 05662 020141         LDA 0,TIME1
12 05663 024216         LDA 1,C1774     ;TIME/1024
13 05664 123700         ANDS 1,0
14 05665 123240         ADDOR 1,0
15 05666 121240         MOVOR 1,0
16 05667 101000         MOV 0,0
17 05670 101405         INC 0,0,SNR
18 05671 000403         JMP .LOP1       ;STALLED 00
19 05672 063400         SKPBN 0         ;MICROSEC
20 05673 000774         JMP .-4
21 05674 062677 .LOP1:  IORST
22 05675 014745         DSZ ITRCT
23 05676 000433         JMP CYCTS       ;NOT 1RQ TIMES ITERATED
24 05677 034742         LDA 3,ITR       ;RESET ITERATION CNTR
25 05700 054742         STA 3,ITRCT
26 05701 034742         LDA     3,ESWIT
27 05702 175405         MOV     3,3,SNR
28 05703 002742 .LOPX:  JMP     @.RTRN
29 05704 054477         READS 3
30 05705 177100         ADDL 3,3
31 05706 177103         ADDL 3,3,SNC
32 05707 000402         JMP PCENT+1
33 05710 006250         PCRLF           ;PRINT CARRIAGE
34 05711 024733         LDA 1,ERRCT
35 05712 030727         LDA 2,ITR
36 05713 004540         JSR MULT
37 05714 030725         LDA 2,ITR
38 05715 004522         JSR DIVD
39 05716 006512         JSR @IPDEC      ;PRINT VALUE
40 05717 020407         LDA 0,PCENT     ;EXAMPLE:  89%
41 05720 006507         JSR @ICHAR
42 05721 063511         SKPBZ TTO
43 05722 000777         JMP .-1
44 05723 063317         SKPBZ LPT
45 05724 000777         JMP .-1
46 05725 000402         JMP PCENT+1
```

```
10085 .MAIN
01
02 05726 000045 PCENT:  "%              ;CHARACTOR
03 05727 176400         SUB     3,3
04 05730 054714         STA     3,ERRCT
05 05731 034712 CYCTS:  LDA 3,ESWIT
06 05732 175224         MOV 3,3,SZR
07 05733 000410         JMP CNS         ;THERE HAS BEEN AN ERROR
08 05734 022714 CYC1:   LDA 0,SAV0      ;RESTORE AC'S
09 05735 024712         LDA 1,SAV1
10 05736 030710         LDA 2,SAV2
11 05737 177112         ADDL# 3,3,SZC   ;SWITCH 1
12 05740 000743         JMP .LOPX       ;(1)=PROCEED
13 05741 006150         JSR @STALL      ;KILL TIME (IORST)
14 05742 002201         JMP @LOOPR      ;(0)=LOOP ROUTINE
15
16 05743 074477 CNS:    READS 3         ;ERROR LOOP, IS A RECAL
17 05744 024051         LDA 1,KBB       ;REQUESTED ?
18 05745 137415         AND# 1,3,SNR
19 05746 000410         JMP CNS1        ;NO  GO-ON
20 05747 175300         MOVS 3,3        ;YES, ASSEMBLE THE
21 05750 024112         LDA 1,C3        ;UNIT # FROM SW6-7
22 05751 137420         ANDZ 1,3
23 05752 021506         LDA 0,TRCL,3
24 05753 040401         STA 0,.+1       ;PICK A RECAL
25 05754 006245         RECL3           ;RECALIBRATE !!
26 05755 004512         JSR .DLY        ;DELAY 1 SEC
27
28 05756 074477 CNS1:   READS 3         ;IF SW9=1 DELAY
29 05757 022060         LDA 0,KS9       ; 1 SEC.
30 05760 117424         AND 0,3,SZR
31 05761 004506         JSR .DLY
32 05762 034467         LDA 3,DSWT      ;FORCED 2 SEC DELAY ??
33 05763 175005         MOV 3,3,SNR
34 05764 000403         JMP .+3         ;NO
35 05765 004502         JSR .DLY        ;YES,   1 SEC
36 05766 004501         JSR .DLY        !       1 SEC
37 05767 074477         READS 3
38 05770 000744         JMP CYC1
39
```

```
10086 .MAIN
01
02 05771 054654 .EH1:   STA 3,.RTRN     ;ERR WITH FORCED 1 SEC DELAY
03 05772 176520         SUBZL 3,3
04 05773 000403         JMP .EH2
05 05774 054651 .EHALT: STA 3,.RTRN     ;ERROR SUBROUTINE
06 05775 176400         SUB 3,3         ;ERROR WITH NO DELAY FORCED
07 05776 054653 .EH2:   STA 3,DSWT      ;DELAY SWITCH
08 05777 034644         LDA 3,ESWIT
09 06000 175004         MOV 3,3,SZR
10 06001 000405         JMP ERET
11 06002 034643         LDA 3,.RTRN
12                      ;ERROR, C(3)=PC
13 06003 063077         HALT            ;HALT TO ALLOW RESETTING
14                      ;SWITCHS
15 06004 054637         STA 3,ESWIT
16 06005 004404         JSR EPRINT
17 06006 010636 ERET:   ISZ ERRCT
18 06007 002636         JMP @.RTRN
19 06010 002635         JMP @.RTRN
20 06011 054635 EPRINT: STA 3,SAV2
21 06012 040636         STA     0,SAV0
22 06013 044634         STA     1,SAV1
23 06014 006260         PCRLF           ;PRINT CARRIAGE
24 06015 006261         MESSAGE         ;AND HEADER
25 06016 006031         HEADER
26 06017 020626         LDA 0,.RTRN
27 06020 040623         STA     0,ESWIT
28 06021 126000         ADC 1,1
29 06022 107400         ADD 0,1
30 06023 004514         JSR PDCT        ;PC OF ERROR
31 06024 020624         LDA     0,SAV0
32 06025 024622         LDA     1,SAV1
33 06026 002620         JMP @SAV2
34 06027 000223 ICHAR:  CHAR
35 06030 006143 IPDEC:  PDEC
36       006031 HEADER: .TXTE 1
37 06031 141520 PC      1
38       002011
```

```
10007 .MAIN
01
02                ;AC1 REM AC0=(AC0,AC1)/AC2
03 00033 102400 DIVIO:  SUB     0,0
04 00034 054431 DIVDO:  STA     3,MSAV
05 00035 142432         SUBZ#   2,0,SZC
06 00036 000413         JMP     DEXT
07 00037 054426 DIVD:   STA 3,MSAV      ;DIVIDE
08 00040 034426         LDA 3,M20
09 00041 125120         MOVZL 1,1
10 00042 101100 DLOOP:  MOVL 0,0
11 00043 142412         SUB# 2,0,SZC
12 00044 142400         SUB 2,0
13 00045 125100         MOVL 1,1
14 00046 175404         INC 3,3,SZR
15 00047 000773         JMP DLOOP
16 00050 175441         SUBO    3,3,SKP
17 00051 176420 DEXT:   SUBZ    3,3
18 00052 000413         JMP @MSAV
19
20                ;(AC0,AC1)=AC1*AC2+AC0
21 00053 102460 MULT:   SUBC 0,0        ;MULTIPLY
22 00054 054411 MULTA:  STA 3,MSAV
23 00055 034411         LDA 3,M20
24 00056 125203 MLOOP:  MOVR 1,1,SNC
25 00057 101201         MOVR 0,0,SKP
26 00060 143220         ADDZR 2,0
27 00061 175404         INC 3,3,SZR
28 00062 000774         JMP MLOOP
29 00063 125240         MOVCR 1,1
30 00064 002401         JMP @MSAV
31 00065 000000 MSAV:   0
32
33 00066 177760 M20:    -20
34
35 00067 020165 .DLY:   LDA 0,C12       ;DELAY 1 SEC
36 00070 040157         STA 0,ITRCNT    ;10,X100MS
37 00071 020142 .DLY1:  LDA 0,TIME
38 00072 060220         NIO 0           ;100 MS TYPE 2
39 00073 064433         DIA 1,DSKP
40 00074 127401         AND 1,1,SKP
41 00075 004406         JSR HL1         ;NO SKP?
42 00076 101404         INC 0,0,SZR     ;STALLED 100?
43 00077 000773         JMP .DLY1+1     ;NOT YET
44 00100 014157         DSZ ITRCNT      ;1 SEC.?
45 00101 000770         JMP .DLY1       ;STALL MORE
46 00102 001400         JMP 0,3
47 00103 063777 HL1:    HALT            ;FATAL ERROR,AC3
48 00104 000777         JMP .-1         ;CONTAINS CALL+1
```

```
10008 .MAIN
01                ;TELETYPE NON INTERRUPT PACKAGE
02                ;AC1,AC2 SAVED
03                ;"MESS" PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLER
04                ;"CRLF" PRINTS A CARRIAGE RETURN
05                ;"POCT" PRINTS C(1) IN OCTAL
06                ;"ZOCT" PRINTS C(1) IN OCTAL, LEADING ZEROS SUPPRESSED
07                ;"POEC" PRINTS C(1) IN DECIMAL, LEADING ZEROS SUPPRESSED,
08                ;THE ABOVE THREE ARE FOLLOWED BY THE TAB IN P.TAB
09                ;"TINO" ACCEPTS OCTAL, AND
10                ;"TINO" ACCEPTS DECIMAL SINGLE PRECISION SIGNED INTEGERS
11                ;INTO AC1 FROM THE TTI.  LEADING NULLS, TABS,
12                ;AND SPACES ARE IGNORED.  A 15 BIT UNSIGNED INTEGER IS
13                ;FORMED, THEN NEGATED IF A MINUS SIGN IS TYPED.
14                ;EXIT AT CALL+1 IF INPUT ERROR WITH AC0=BAD CHARACTER,
15                ; (NOT A LEGAL DIGIT OR TERMINATING CHARACTER)
16                ;EXIT AT CALL+2 UPON TERMINATING CHARACTER
17                ; WITH AC0=0, 0, 40, 12, 55
18                ; FOR NULL, TAB, SPACE, CARRIAGE RETURN, COMMA
19                ;THE ABOVE WAIT FOR TTO DONE, THEN CLEAR TTO.
20                ;"CHAR" PRINTS ASCII CHARACTOR IN C(0)R; C(0)L MUST BE 0.
21                ;EXITS CALL +2 IF C(0)R=0, CORRECTS THE PARITY.
22                ;SIMULATES TAB ON ASR33.
23                ;"TYPE" PRINTS C(0)R.  MUST HAVE PROPER PARITY.  EXITS AT
24                ;CALL+1.  REPLACE "TYPE" WITH INTERRUPT TYPE IF DESIRED.
25
26 00105 054554 MESS:   STA 3,MESSR     ;PRINT A TEXT MESSAGE
27 00106 044510         STA     1,P.AC1
28 00107 050510         STA     2,P.AC2
29 00110 010551         ISZ MESSR
30 00111 031400         LDA 2,0,3       ;C(2) POINTS TO MESSAGE
31 00112 024510         LDA 1,P,377     ;A 8 BIT MASK
32 00113 021000         LDA 0,0,2       ;C(2)=DATA WORD
33 00114 125112         MOVLW 1,1,SZC
34 00115 123701         ANDS 1,0,SKP
35 00116 123401         AND 1,0,SKP     ;C(0)=DATA CHARACTOR RIGHT
36 00117 151400         INC 2,2         ;INC TO NEXT WORD
37 00120 124000         COM 1,1         ;FLIP MASK
38 00121 004302         JSR CHAR        ;PRINT
39 00122 000771         JMP MESS+6      ;ANOTHER
40 00123 000402         JMP     .+2
41 00124 004477 P.LST:  JSR     CHAR
42 00125 024471 PEXIT:  LDA     1,P.AC1
43 00126 030471         LDA     2,P.AC2
44 00127 063511         SKPBZ TTO
45 00130 000777         JMP .-1
46 00131 060211         NIOC TTO
47 00132 063517         SKPBZ LPT
48 00133 000777         JMP .-1
49 00134 060217         NIOC LPT
50 00135 002524         JMP @MESSR      ;LAST
```

```
10089 .MAIN
01 00136 102401 IOCT:   SUB 0,0,SKP
02 00137 020462 POCT:   LDA     0,P.C60
03 00140 052457         STA     2,P.AC2
04 00141 030435         LDA 2,OCTAB     ;PRINT C(1) IN OCTAL
05 00142 000424         JMP .+4
06 00143 052454 PDEC:   STA     2,P.AC2
07 00144 030442         LDA 2,DECTB     ;PRINT C(1) IN DECIMAL
08 00145 102400         SUB 0,0
09 00146 054513         STA 3,MESSR     ;BOTH ENTRYS PRINT NUMBER
10 00147 044447         STA     1,P.AC1
11 00150 040445         STA 0,ZSUPP     ;THEN TAB TO NEXT POSITION
12 00151 050431         STA 2,.+1
13 00152 000000 DECOCT: 0               ;A"LDA 2,TABLE" INSTRUCTION
14 00153 010777         ISZ .-1
15 00154 020444         LDA 0,P.TAB
16 00155 151005         MOV 2,2,SNR     ;IF TABLE ENTRY=0
17 00156 000747         JMP PEXIT       ;EXIT WITH NO TAB
18 00157 034436         LDA 3,ZSUPP     ;ZEROS SUPPRESS STUF
19 00160 102400         SUB 0,0
20 00161 146452 DECOT:  SUBOM 2,1,SZC
21 00162 000405         JMP DECP
22 00163 146400         SUB 2,1         ;FORM THE DIGIT
23 00164 034435         LDA 3,P.C60
24 00165 102420         INC 0,0
25 00166 000773         JMP DECOT
26 00167 151235 DECP:   MOVZM 2,2,SNR
27 00170 034431         LDA 3,P.C60
28 00171 054424         STA 3,ZSUPP     ;C(0)=DIGIT
29 00172 103000         ADD 3,0         ;MAKE ASCII
30 00173 175004         MOV 3,3,SZR
31 00174 004427         JSR CHAR        ;PRINT
32 00175 000755         JMP DECOCT      ;GET NEXT DIGIT
33
34 00176 030425 OCTAB:  LDA 2,.+1+.-DECOCT
35 00177 100000         100000
36 00200 010000         10000
37 00201 001000         1000
38 00202 000100         100
39 00203 000010         10
40 00204 000001         1
41 00205 000000         0
42 00206 030435 DECTB:  LDA 2,.+1+.-DECOCT
43              000012  .RDX 10
44 00207 023420         10000
45 00210 001750         1000
46 00211 000144         100
47 00212 000012         10
48 00213 000001         1
49 00214 000000         0
50              000010  .RDX 8
51 00215 000000 ZSUPP:  0
52 00216 000000 P.AC1:  0
53 00217 000000 P.AC2:  0
54 00220 000011 P.TAB:  11      ;CHARACTER PRINTED AFTER NUMBERS
55 00221 000060 P.C60:  60
56 00222 000377 P.377:  377
```

```
10090 .MAIN
01 00223 054434 CHAR:   STA 3,CHRET     ;PRINT C(0) RIGHT
02 00224 101305         MOVS 0,0,SNR    ;RETURN +2 IF NULL
03 00225 001401         JMP 1,3
04 00226 115120         MOVZL   0,3     ;COMPUTE EVEN PARITY
05 00227 177004         ADD     3,3,SZR
06 00230 000777         JMP     .-1
07 00231 103200         ADDR    0,0
08 00232 101300         MOVS    0,0
09 00233 034403 CHAR1:  LDA     3,P.C11 ;IS THIS A TAB
10 00234 116415         SUBM 0,3,SNR
11 00235 000403         JMP CHA.3               ;YES
12 00236 004424         JSR TYPE        ;NO PRINT IT
13 00237 002420         JMP @CHRET      ;EXIT
14 00240 020401 CHA.3:  LDA     0,P.240 ;SIMULATE A TAB
15 00241 004421         JSR     TYPE    ;WITH 1 TO 7 SPACES
16 00242 020416         LDA     0,CHORZ
17 00243 034452         LDA     3,P.C7
18 00244 153404         AND     3,3,SZR
19 00245 000773         JMP     CHA.3
20 00246 040412         STA 0,CHORZ
21 00247 002410         JMP @CHRET
22
23 00250 054411 CRLF:   STA 3,MESSR     ;SAVE RETURN
24 00251 044745         STA     1,P.AC1
25 00252 030745         STA     2,P.AC2
26 00253 020445         LDA 0,P.C15
27 00254 004747         JSR CHAR        ;PRINT CARRIAGE AND LF
28 00255 020442         LDA     0,P.C12
29 00256 000346         JMP     P.LST
30
31 00257 000000 CHRET:  0
32 00260 000000 CHORZ:  0
33 00261 000000 MESSR:  0
```

```
;#091 .MAIN
#1 06242 054441 TYPE1:   STA 3,TYPRET     ;TYPE THE C(#)R IF
#2 06243 074477          READS 3          ;SWITCH 2(#).
#3 06244 175100          MOVL 3,3
#4 06245 177122          ADDZL 3,3,SZC
#5 06246 000420          JMP TYPE2
#6 06247 063511          SKPBZ TTO
#7 06250 000777          JMP .-1
#8 06251 061111          DOAS #,TTO
#9 06252 000414          JMP TYPE2
10 06253 034727 TYPE1:   LDA     3,P.377
11 06254 175220          MOVZR   3,3
12 06255 163400          AND     3,#
13 06256 116043          ADCO    #,3,SNC
14 06257 034423          LDA     3,P.C48
15 06300 102432          SUBZ#   3,#,SZC ;SKIP NON-PRINTING CHAR
16 06301 012757          ISZ     CHORZ
17 06302 034416          LDA     3,P.C15
18 06303 116445          SUBO    #,3,SNR
19 06304 054754          STA     3,CHORZ ;CLR HORZ POS
20 06305 002416          JMP #TYPRET
21 06306 175100 TYPE2:   MOVL 3,3
22 06307 177103          ADDL 3,3 SNC
23 06310 000763          JMP TYPE1
24 06311 063517          SKPBZ LPT
25 06312 000777          JMP .-1
26 06313 061117          DOAS #,LPT
27 06314 000757          JMP TYPE1
28 06315 000007 P.C7:    7
29 06316 000011 P.C11:   11
30 06317 000012 P.C12:   12
31 06320 000015 P.C15:   15
32 06321 000240 P.240:   240
33 06322 000040 P.C40:   40
34 06323 000000 TYPRET:  0
```

```
;#092 .MAIN
#1 06324 020773 TINC.:   LDA     #,P.C12
#2 06325 004735          JSR     TYPE
#3 06326 010733 TINX.:   ISZ     MESSR
#4 06327 024667 TINR.:   LDA     1,P.AC1
#5 06330 034665          LDA     3,ZSUPP
#6 06331 175102          MOVL    3,3,SZC
#7 06332 124400          NEG     1,1
#8 06333 002276          JMP     #PEX
#9
10 06334 102121 TIND:    ADCZL   #,#,SKP ;OCTAL ENTRY
11 06335 102440 TIND:    SUBO    #,#     ;DECIMAL ENTRY
12 06336 054723          STA     3,MESSR
13 06337 050630          STA     2,P.AC2 ;AC2 IS SAVED
14 06340 030757          LDA     2,P.C12
15 06341 113000          ADD     #,2
16 06342 102440          SUBO    #,#
17 06343 040652          STA     #,ZSUPP ;MINUS SIGN AND LEADING SPACES FLAG
18 06344 034651 TINS.:   LDA     3,ZSUPP
19 06345 175004          MOV     3,3,SZR
20 06346 000760          JMP     TINX.
21 06347 054647 TINN.:   STA     3,P.AC1
22 06350 063610          SKPDN   TTI
23 06351 000777          JMP     .-1
24 06352 060610          DIAC    #,TTI
25 06353 004650          JSR     CHAR
26 06354 034748          LDA     3,P.C48
27 06355 116414          SUB#    #,3,SZR
28 06356 101015          MOV#    #,#,SNR
29 06357 000765          JMP     TINS.   ;SPACE, TAB, OR NULL
30 06360 024432          LDA     1,TIN2.
31 06361 106015          ADC#    #,1,SNR ;COMMA
32 06362 000744          JMP     TINX.
33 06363 106424          SUBZ    #,1,SZR ;MINUS
34 06364 000405          JMP     TINM.   ;NO
35 06365 034630          LDA     3,ZSUPP
36 06366 177200          ADDR    3,3     ;COMPLEMENT SIGN
37 06367 054626          STA     3,ZSUPP
38 06370 000760          JMP     TINN.+1
39 06371 136415 TINM.:   SUB#    1,3,SNR ;IS IT A CARRIAGE RETURN?
40 06372 000732          JMP     TINC.
41 06373 024416 TINN.:   LDA     1,TIN1.
42 06374 107022          ADDZ    #,1,SZC ;SKIP IF NOT A DIGIT
43 06375 146513          SUBL#   2,1,SNC ;SKIP IF DIGIT
44 06376 000731          JMP     TINR.
45 06377 010616          ISZ     ZSUPP   ;OUT OF LEADING SPACES
46 06400 020516          LDA     #,P.AC1
47 06401 101120          MOVZL   #,#
48 06402 115120          MOVZL   #,3
49 06403 175120          MOVZL   3,3
50 06404 137000          ADD     1,3     ;# OLD P.AC1'S + NEW DIGIT
51 06405 145220          MOVZR   2,1
52 06406 125202          MOVZR#  1,1,SZC ;SKIP IF OCTAL MODE
53 06407 117300          ADD     #,3     ;ADD 2 OLD P.AC1'S
54 06410 000737          JMP     TINM.
55 06411 177720 TIN1.:   -50
56 06412 000055 TIN2.:   55
```

```
10093 .MAIN
01 06413 054425 DEVC:   STA 3,SV3
02 06414 026260         PCRLF
03 06415 025261         MESSAGE
04 06416 026573         MSG7
05 06417 026275 ACC:    JSR @ITIN
06 06420 000427         JMP EK
07 06421 030171         LDA 2,C33
08 06422 132415         SUBN1,2 SNR
09 06423 000418         JMP I33
10 06424 030272         LDA 2,C73
11 06425 132415         SUBN1,2 SNR
12 06426 000417         JMP I73
13 06427 026260 ER:     PCRLF
14 06430 006261         MESSAGE
15 06431 006611         MSG8
16 06432 000765         JMP ACC
17 06433 030171 I33:    LDA 2,C33
18 06434 145000         MOV 2,1
19 06435 002403         JMP @SV3
20 06436 030220 I73:    LDA 2,C73
21 06437 000775         JMP I33+1
22 06440 000000 SV3:    0
23 06441 005215 MSG8:   .TXTE |<15><12>MOUNT SCRATCH DISK,SET DATA SWITCHES,<15>
24       147515
25       047125
26       120324
27       141523
28       040722
29       141724
30       122110
31       144504
32       045523
33       051654
34       152305
35       042240
36       150101
37       120101
38       153523
39       152311
40       044303
41       051765
42       106654
43 06465 050012 PRESS CONTINUE<15><12>|
44       142722
45       051523
46       141040
47       047317
48       144724
49       052516
50       120725
51       000012
52 06476 054724 MSG1:   .TXTE |TYPE 0 FOR CARTRIDGE|
53       142520
54       030240
55       143240
56       151317
57       141540
58       151101
59       151324
60       042311
```

```
 0094 .MAIN
01       142507
02       000000
03 06511 054724 MSG2:   .TXTE |TYPE 1 FOR 10 SURFACE DISK|
04       142520
05       130540
06       143240
07       151317
08       130040
09       120060
10       052523
11       143322
12       141501
13       120305
14       144504
15       045523
16       000000
17 06527 054724 MSG3:   .TXTE |TYPE 2 FOR 20 SURFACE DISK|
18       142520
19       131240
20       143240
21       151317
22       131240
23       120060
24       052523
25       143322
26       141501
27       120305
28       144504
29       045523
30       000000
31 06545 020540 MSG4:   .TXTE | -7- |
32       026477
33       000240
34 06558 054724 MSG5:   .TXTE |TYPE UNIT NUMBERS (0-3) TO TEST|
35       142520
36       052040
37       144516
38       120324
39       052516
40       041115
41       151305
42       120123
43       030050
44       031455
45       120251
46       147724
47       152240
48       051705
49       000324
50 06570 040520 MSG6:   .TXTE |PASS|
51       051523
52       000000
53 06573 047305 MSG7:   .TXTE |ENTER DEVICE CODE(33 OR 73)|
54       142724
55       120322
56       142504
57       144520
58       142703
59       141540
60       042317
```

```
 0095 .MAIN
01        024305
02        031463
03        147640
04        120322
05        031667
06        000251
07 00611  047311 MSG8:   .TXTE [INVALID ENTRY, ENTER 33 OR 73]
08        040526
09        144714
10        120104
11        047305
12        151324
13        126131
14        142640
15        152110
16        151305
17        031640
18        120063
19        151317
20        133640
21        000063
22 00638  000000 PRGEN:  0
23 00631  047503         .TXT   /COPYRIGHT (C) DGC 1971,72,73,74,75,76
24        054520
25        044522
26        044107
27        020124
28        041450
29        020051
30        043504
31        020103
32        034461
33        030467
34        033454
35        026062
36        031467
37        033454
38        026064
39        032467
40        033454
41 00653  040460 ALL RIGHTS RESERVED/
42        046114
43        051040
44        043511
45        052110
46        020123
47        042522
48        042523
49        053122
50        042125
51        000000
52 00666  045504 DIRT:   .TXTE [DKP DIAG 10]
53        120120
54        144504
55        043501
56        130640
57        000060
58 00674  000002         000002
```

```
 0096 .MAIN
01 00675  000200         DTOSB
02 00676  000001         000001
03 00677  070707         70707
04 00700  070707         70707
05 00701  070707         70707
06 00702  070707         70707
07 00703  100033         100033
08
09
```

```
10097 .MAIN
01              ;PROCESSOR TIMER PACKAGE
02              ;THIS PACKAGE IS CALLED WHENEVER IT IS NECESSARY TO
03              ;IDENTIFY THE MEAN TIME BASE OF THE COMPUTER IN
04              ;WHICH THE PROGRAM RESIDES. THE MEAN TIME BASE MAY
05              ;THEN BE UTILIZED TO VERIFY OR CALCULATE THE RELATIONSHIPS
06              ;OF OTHER PERIPHERAL FUNCTIONS.
07              ;
08              ;THE PACKAGE RETURNS TO THE CALL INSTRUCTION WITH
09              ;THE CONTENTS OF AC1= TO A CALIBRATION COUNT
10              ;THAT MAY BE INCREMENTED TO OVERFLOW IN 100 MILLI-
11              ;SECONDS BY THE FOLLOWING DELAY LOOP.
12              ;TYPE1: MOV 0,0
13              ;       INC 0,0,SZC     ;SKP=NOT OVERFLOW
14              ;       JMP 1,3         ;EXIT LOOP
15              ;       0               ;ANY FLAVOR IO SKP
16              ;       JMP TYPE1
17              ;
18              ;THE CONTENTS OF AC2 WILL CONTAIN A SIMILAR 100 MS
19              ;ITERATION COUNT BUT FOR THE FOLLOWING LOOP:
20              ;TYPE2: NIO 0
21              ;       DIA 1,.DEV      ;GET DEVICE STATUS
22              ;       ANDM 2,1,SZR    ;ANY STATUS COMPARE
23              ;       JMP .+4         ;EXPECTED STATUS EXIT
24              ;       INC 0,0,SZR     ;SKP OUT ON LOOP OFLOW
25              ;       JMP TYPE2
26              ;
27              ;THE VALUES RETURNED MAY BE ARITHMETICALLY
28              ;PROCESSED (MULTIPLIED/DIVIDED) FOR LONGER OR
29              ;SHORTER DELAYS AS LONG AS THE STANDARD LOOPS
30              ;LISTED ABOVE ARE UTILIZED.
31              ;
32              ;IT IS RECOMMENDED THAT ALL TIMING FUNCTIONS
33              ;BE PERFORMED WITHIN THE SAME GENERAL AREA OF
34              ;MEMORY AS THIS TIMING PACKAGE.
```

```
10098 .MAIN
01              ;
02              ;INITIALLY, THE TIMER PACKAGE ATTEMPTS TO
03              ;DETERMINE IF THE COMPUTER HAS A REAL TIME CLOCK
04              ;AVAILABLE. THE ITERATION COUNTS ARE DEVELOPED SIMPLY
05              ;BY SYNCING WITH THE CLOCK AND COUNTING
06              ;THE #OF LOOP ITERATIONS AT 10 HZ.
07              ;
08              ;IF, HOWEVER THERE IS NO REAL TIME CLOCK THE MEAN
09              ;TIME BASE OF THE LOOPS MUST BE CALCULATED. THIS
10              ;IS PERFORMED BY COUNTING THE #OF TIMES THE
11              ;STANDARD LOOPS ARE ITERATED FOR ONE OUTPUT
12              ;CHARACTER TO DEVICE "TTO" AND REQUESTING THE
13              ;BAUD RATE OF DEVICE TTO BE TYPED IN BY THE
14              ;TEST OPERATOR.
15              ;
16 06704 054274 PTIME:  STA 3,SVTIME
17 06705 063514         SKPBZ RTC       ;TEST FOR CAS/RTC
18 06706 000416         JMP SCORE       ;CAS RTC NONEXIS.
19 06707 060114         NIOS RTC        ;TURN CLOCK ON
20 06710 063514         SKPBZ RTC       ;BUSY =1 IS RTC
21 06711 000403         JMP .+3
22 06712 063514         SKPDN RTC       ;DONE =0 NO RTC
23 06713 000411         JMP SCORE       ;AND DEV TTO IS USED
24 06714 052677         IORST
25 06715 102520         SUBZL 0,0       ;=10 HZ FOR RTC
26 06716 004473         JSR TYME
27 06717 061114         DOAS 0,RTC      ;PASSED TO "TYME"
28 06720 063314         SKPBZ RTC       ;FOR EXECUTION
29 06721 002274         JMP @SVTIME     ;AC1 AND AC2=LOOP COUNTS
30              ;
31              ;THE FOLLOWING SUBROUTINE IS TO RETAIN COMPATABILITY
32              ;WITH THE OLD PROCESSOR IDENTIFICATION AND TIMING
33              ;PACKAGE TO RETRIEVE THE LOOP COUNT FOR
34              ;THE DIA,B OR C LOOP TYPE 2
35 06722 024542 .301:   LDA 1,NUCAL
36 06723 001400         JMP 0,3
```

```
00000 .MAIN
00:                     ;
00                      ;THERE IS NO RTC-UTILIZE DEVICE TTO AND ASK FOR
00                      ;BAUD RATE INPUT FROM OPERATOR
00 06724 062677 SCORE:  IORST
00 06725 102400         SUB 0,0             ;AC0=NULL CHARACTER
00 06726 004463         JSR TYME
00 06727 061111         DOAS 0,TTO          ;PASSED TO TYME
00 06730 063511         SKPBZ TTO           ;FOR EXECUTION
00 06731 006534 SCORA1: JSR @TUMBLER        ;OUT TEXT
00 06732 007075         SESOUT              ;ASKING FOR BAUD RATE
00                      ;THE FOLLOWING SERIES OF INSTRUCTIONS WILL
00                      ;CALCULATE THE ITERATION COUNT FOR
00                      ;1 BIT OF TTO OUTPUT AFTER RETRIEVING
00                      ;THE CONSOLE BAUD RATE FROM THE
00                      ;TEST OPERATOR---REQUIRES SUBROUTINE TIND
00 06733 006534         JSR @KEYS
00 06734 007775         JMP SCORA           ;INPUT ERROR
00 06735 044531         STA 1,LOCK          ;SAVE BAUD RATE
00 06736 030536         LDA 2,3,3D1         ;10
00                      ;ROUTINE ASSUMES AN 11 BIT CHARACTER
00 06737 151400         INC 2,2             ;ASSUME 11 BITS
00 06740 024523         LDA 1,ORDINAL       ;COUNT FOR FULL CHAR
00 06741 102400         SUB 0,0
00 06742 006527         JSR @KEYS+2         ;CHAR TIME/#BITS
00 06743 101004         MOV 0,0,SZR         ;IF ANY REM,
00 06744 125420         INC 1,1             ;FUDGE BIT COUNT
00 06745 022521         LDA 0,LOCK
00 06746 044520         STA 1,LOCK          ;SAVE ITR COUNT 1 BIT
00 06747 131000         MOV 1,2
00 06750 125000         MOV 0,1             ;AC1 = BAUD RATE
00                      ;BAUD RATE TIMES COUNT FOR 1 BIT
00                      ;WILL EQUAL ITERATION COUNT FOR 1 SECOND
00 06751 102400         SUB 0,0
00 06752 006516         JSR @KEYS+1         ;MUL AC1*AC2
00 06753 040520         STA 0,KN            ;SAVE DOUBLE LENGTH
00 06754 044516         STA 1,KS            ;RESULT
00                      ;1 SECOND DIVIDED BY 10 = 100 MILLISECONDS
00 06755 030517         LDA 2,3,3D1         ;10
00 06756 006513         JSR @KEYS+2
00 06757 030524         LDA 2,ORDINAL       ;COUNT FOR 1 CHAR
00 06760 044503         STA 1,ORDINAL       ;ORDINAL=100 MS TYPE 1
00                      ;CONTINUE CALCULATIONS NEXT PAGE
```

```
10100 .MAIN
01                      ;
02                      ;1 SECOND COUNT/1 CHAR COUNT = # CHARS PER SEC
03                      ;THIS CALCULATION IS USED TO EXPAND THE TYPE 2 COUNT
04 06761 020512         LDA 0,KN
05 06762 024510         LDA 1,KS            ;1 SEC. RESTORED
06 06763 006506         JSR @KEYS+2         ;DIVIDE BY CHAR.
07 06764 044506         STA 1,KS            ;# CHAR. 1 SEC
08                      ;CALC RELATIONSHIP OF REM. TO 1 CHAR TO FILL SECOND
09 06765 145120         MOVZL 2,1
10 06766 111005         MOV 0,2,SNR
11 06767 151400         INC 2,2
12 06770 102400         SUB 0,0
13 06771 006500         JSR @KEYS+2         ;DIVIDE REM INTO CHAR
14                      ;AC1=FUDGE FACTOR 1 RELEATIONSHIP OF CHAR TO TOTAL 1 SEC
15                      ;FINISH CALCULATIONS ON LOOP TYPE 1 TO= 1SECOND
16 06772 131000         MOV 1,2             ;FUDGE FACTOR
17 06773 024471         LDA 1,NUCAL         ;INTO CHARACTER TIME
18 06774 125120         MOVZL 1,1
19 06775 102400         SUB 0,0             ;WILL =
20 06776 006473         JSR @KEYS+2         ;PORTION OF CHAR
21 06777 121000         MOV 1,0             ;TO COMPLETE 1 SECOND
22 07000 024464         LDA 1,NUCAL         ;1 CHAR, TYPE 2 LOOP
23 07001 030471         LDA 2,KS            ;# CHARS IN 1 SEC
24 07002 006466         JSR @KEYS+1         ;+ PORTION OF CHAR
25                      ;DOUBLE LENGTH AC0,AC1=1 SECOND FOR TYPE 2 LOOP
26 07003 030471         LDA 2,3,3D1         ;DIVIDE BY 10 FOR 100 MS
27 07004 006465         JSR @KEYS+2
28 07005 044457         STA 1,NUCAL
29 07006 131000         MOV 1,2             ;AC2=100MS LOOP2
30 07007 024454         LDA 1,ORDINAL       ;AC1 =100MS LOOP 1
31 07010 002274         JMP @SVTIME
```

.END

0104 .MAIN

A0	000503	9/07	68/06			
A1	000525	10/02				
A10	000602	11/10				
A11	000611	11/18				
A12	000620	11/26				
A13	000630	11/35				
A14.2	000642	12/06	12/15			
A2	000531	10/07				
A3	000535	10/12				
A30.4	000656	12/21	12/30			
A4	000541	10/17				
A46.6	000672	13/06	13/15			
A5	000345	10/22				
A6	000552	10/28				
A62.7	000706	13/21	13/30			
A7	000557	10/34				
A78	000720	14/02				
A79	000731	14/12				
A8	000364	10/40				
A80	000741	14/21				
A81	000754	15/02				
A82	000767	15/14				
A83	001002	15/26				
A84	001015	15/38				
A85	001030	16/02				
A86	001043	16/14				
A87	001056	16/26				
A9	000573	11/02				
ACC	006417	93/05	93/16			
ADR	004224	57/27	58/28	58/40		
ADR1	004337	59/03	60/03	60/42		
ALT0	005206	47/09	74/15			
ALT1	005204	46/37	74/12			
B1	001071	16/38				
B10	001210	18/22				
B11	001224	18/35				
B12	001240	19/02				
B13	001254	19/15				
B14	001265	19/25				
B15	001276	19/35				
B16	001307	20/02				
B17	001320	20/12				
B18	001337	20/14	20/28			
B19	001357	20/31	21/01			
B1.1	001075	16/39	16/42	16/47	16/48	16/55
B1.2	001102	16/47	16/54			
B2	001113	16/51	17/02			
B20	001377	21/04	21/18			
B21	001417	21/21	21/35			
B22	001433	21/41	21/49			
B23	001445	21/52	22/02			
B24	001457	22/05	22/13			
B25	001471	22/16	22/24			
B26	001507	22/30	22/40			
B27	001523	22/43	23/02			
B28	001537	23/05	23/15			
B29	001553	23/18	23/28			
B3	001121	17/09				
B30	001566	23/30	23/40			

```
10101 .MAIN
01                      ;
02                      ;THE FOLLOWING SUBROUTINE DETERMINES THE ITERATION
03                      ;COUNTS FOR THE DEVICE SPECIFIED BY THE INSTRUCTIONS
04                      ;FOLLOWING THE JSR CALL TO TYMF
05                      ;DOAS 0,RTC     OR      DOAS 0,TTO
06                      ;SKPBZ RTC      OR      SKPBZ TTO
07                      ;
08 07011 024001 TYME:   LDA 1,1         ;SAVE INTR. LINK
09 07012 044450         STA 1,RVTMP
10 07013 024437         LDA 1,ENTYM
11 07014 044001         STA 1,1         ;FOR LOOP 2 INTR.
12 07015 025400         LDA 1,0,3       ;GET DOAS
13 07016 044410         STA 1,TIMA
14 07017 044414         STA 1,TIMB
15 07020 044421         STA 1,TIMC      ;FOR EXECUTE
16 07021 025401         LDA 1,1,3       ;GET SKPBZ
17 07022 044405         STA 1,TIMA+1
18 07023 044414         STA 1,TIMB+4    ;FOR EXECUTE
19 07024 152420         SUB 2,2
20 07025 126400         SUB 1,1         ;CLR CTRS
21 07026 061114 TIMA:   DOAS 0,RTC      ;OR TTO
22 07027 063514         SKPBZ RTC
23 07030 000777         JMP .-1         ;WAIT FOR DONE
24 07031 124004         COM 1,1,SZR     ;AND 2ND DONE
25 07032 000774         JMP TIMA        ;THEN START COUNTING
26 07033 061114 TIMB:   DOAS 0,RTC      ;THE THIRD DONE
27                      ;THE FOLLOWING COMPRISES LOOP TYPE 1
28 07034 101000         MOV 0,0
29 07035 125405         INC 1,1,SNR     ;WATCH FOR OFLOW
30 07036 000403         JMP .+3
31 07037 063514         SKPBZ RTC
32 07040 000774         JMP TIMB+1
33                      ;LOOP TYPE 2 IS COUNTED UNTIL PI FROM DEVICE
34 07041 061114 TIMC:   DOAS 0,RTC
35 07042 060177         INTEN
36                      ;THE FOLLOWING INSTR. COMPRISE THE LOOP TYPE 2
37 07043 060000         NIO 0   ;AND IT ITERATES UNTIL
38 07044 060420         DIA 0,0 ;INTERRUPTED BY PI
39 07045 102005         ADC 0,0,SNR
40 07046 004412         JSR HL          ;FILL INSTR.
41 07047 151424         INC 2,2,SZR     ;LOOP CTR
42 07050 000773         JMP TIMC+2
43 07051 004427         JSR HL  ;DEVICE OR PI FAILED
44 07052 077053 ENTYM:  .+1     ;TO HERE WHEN PI
45 07053 044410         STA 1,ORDINAL   ;SAVE LOOP 1
46 07054 054411         STA 2,NUCAL     ;AND LOOP 2
47 07055 022425         LDA 0,RVTMP
48 07056 040001         STA 0,1         ;RESTORE INTR. LINK
49 07057 001402         JMP 2,3 ;RETURN TO CALL
50 07060 063777 HL:     HALT            ;FATAL ERROR,AC3
51 07061 000777         JMP .-1         ;CONTAINS CALL+1
```

```
10102 .MAIN
01                      ;
02                      ;CONSTANTS SUBR. LINKS AND TEMP STORES
03 07062 000000 RVTMP:  0
04 07063 000000 ORDINAL:        0
05 07064 000000 NUCAL:  0
06 07065 006105 TUMBLER:        MESS
07 07066 000000 LOCK:   0
08 07067 006335 KEYS:   TIND
09 07070 006054         MULTA
10 07071 006034         DIVDO
11 07072 000000 KS:     0
12 07073 000000 KN:     0
13 07074 000012 S.3D1:  10.
14 07075 005215 SESOUT: .TXTE 1<15><12><12>
15 07076 152012 TTO BAUD RATE ?= 1
16       147724
17       041240
18       052501
19       120104
20       040722
21       142724
22       037640
23       120275
24       000000
```

```
        007110          .LOC    7110    ;TABLE WITH WRITE BUFFER ADDRESSES.
 007110 000153          153
 007111 003364          3364
 007112 003366          3366
 007113 003400          3400
 007114 003413          3413
 007115 003415          3415
 007116 003427          3427
 007117 003442          3442
 007120 003444          3444
 007121 003456          3456
 007122 003471          3471
 007123 003473          3473
 007124 003505          3505
 007125 003520          3520
 007126 003522          3522
 007127 003534          3534
 007130 003553          3553
 007131 003555          3555
 007132 003567          3567
 007133 003604          3604
 007134 003606          3606
 007135 003630          3630
 007136 003657          3657
 007137 003661          3661
 007140 003703          3703
 007141 003735          3735
 007142 003737          3737
 007143 003771          3771
 007144 004036          4036
 007145 004040          4040
 007146 004075          4075
 007147 004132          4132
 007150 004134          4134
 007151 004172          4172
 007152 004252          4252
 007153 004254          4254
 007154 004313          4313
 007155 004415          4415
 007156 004422          4422
 007157 004430          4430
 007160 004437          4437
 007161 004456          4456
 007162 004460          4460
 007163 004472          4472
 007164 004477          4477
 007165 004541          4541
 007166 004560          4560
 007167 005520          5520
 007170 005522          5522
 007171 005534          5534
 007172 000000          0
```

```
07173 003373          3373            ;READ BUFFER ADDRESSES.
07174 003401          3401
07175 003422          3422
07176 003430          3430
07177 003451          3451
07200 003457          3457
07201 003500          3500
07202 003506          3506
07203 003527          3527
07204 003535          3535
07205 003562          3562
07206 003570          3570
07207 004432          4432
07210 004440          4440
07211 004465          4465
07212 004500          4500
07213 005527          5527
07214 005535          5535
07215 000000          0               ;END OF TABLE.

                      .END
```

B31	001622	23/43	24/02					
B32	001616	24/05	24/15					
B35	001632	24/18	24/28					
B36	001644	24/35						
B37	001646	25/02						
B38	001654	25/09						
B39	001662	25/16						
B4	000130	17/17						
B40	001672	25/18	25/25					
B41	001703	25/28	25/35					
B42	001714	25/38	26/02					
B43	001725	26/05	26/12					
B44	001746	26/24						
B45	001753	26/36						
B46	001765	26/47						
B47	001777	27/02						
B48	002012	27/14						
B49	002023	27/24						
B5	001137	17/25						
B50	002036	27/36						
B51	002051	28/02						
B52	002065	28/15						
B53	002076	28/04	28/26					
B54	002113	28/40						
B55	002124	28/29	29/02					
B56	002141	29/16						
B57	002152	29/05	29/26					
B58	002167	29/40						
B6	001145	17/32						
B7	001156	17/42						
B8	001166	18/02						
B9	001176	18/11						
BEGIN	000230	7/15	7/42					
BGNAD	000222	7/13	7/15					
BIT1	000127	6/16	14/23	15/40				
BIT2	000130	6/17	15/04	16/04				
BIT4	000131	6/18	15/16	16/16				
BIT8	000132	6/19	15/28	16/28				
BUFF	000153	6/37	38/05	38/24	37/03	37/16	37/32	38/03
		38/19	38/36	38/47	39/03	39/22	39/41	40/03
		40/25	41/03	41/11	41/33	42/10	64/34	64/49
		57/03						
C017	000125	6/13	35/21					
C037	000126	6/14	35/37	39/14	39/33	40/17	40/40	
C1000	000263	7/39	17/03	17/18	17/18	17/43	18/03	18/12
		18/25	18/38	19/05	78/17			
C11	000154	7/05	40/07					
C12	000155	7/26	87/35					
C120	002234	7/19						
C137	000237	7/28	54/03	55/03				
C1400	000222	7/33	79/12					
C140K	002451	7/37	16/03	78/09	79/18			
C15	000156	7/07	22/25	76/19	79/15			
C157	002210	7/21	52/50	53/28				
C16	002167	7/08	39/25	39/44				
C17	000114	6/04	32/21	37/05	37/19	37/35	38/06	38/22
		38/39	39/06	43/24	43/28	43/37	44/04	52/27
		53/04	57/24	58/04				

C174K	000226	7/38	78/27					
C177	000117	6/07	33/21	75/32	76/17			
C1774	000216	7/27	59/11	60/11	84/12			
C1777	000122	6/10	34/21					
C2	000010	4/12	63/47					
C204H	000224	7/35	43/47					
C2525	000133	6/21	16/40	38/26	74/12			
C277	000211	7/22	54/05	55/05				
C3	000112	6/02	25/11	31/37	40/09	74/06	75/39	76/27
		85/21						
C30	000170	7/09	24/37					
C312	000212	7/23	58/22	58/58	51/02	51/38	64/04	64/13
		65/10						
C317	000213	7/24	52/52	53/30				
C33	000171	7/10	7/46	77/06	93/07	93/17		
C36	000172	7/11	61/31					
C3600	000223	7/34	17/27	17/37	18/16	18/28	18/41	19/08
C37	000115	6/05	32/37					
C377	000120	6/08	33/37	65/11	73/23	74/27		
C3777	000123	6/11	34/37					
C4	000161	7/02	66/34	74/89				
C402	000214	7/25	38/48					
C420	000215	7/26	46/19					
C5	000162	7/03	21/36	26/17	26/29	73/31	81/59	85/07
C500	004572	64/16	64/54	65/24				
C5252	000134	6/22	74/15					
C6	000163	7/04						
C60	000203	7/16	75/34	76/22				
C63	000204	7/17	75/35	76/23				
C7	000113	6/03	32/05					
C70	000205	7/18	44/12					
C73	000220	7/31	93/10	93/20				
C74K	000225	7/36	28/22	28/39	21/12	21/29	26/19	73/36
C77	000116	6/06	16/49	33/05				
C777	000121	6/29	34/05	71/27				
C7777	000124	6/12	35/05					
CCNT	004573	64/17	64/55	65/16	65/25			
CDBK	000147	6/33	16/52	27/08	77/08	77/17		
CHAR	006223	86/34	88/38	89/41	89/31	90/01	90/27	92/25
CHAR1	006233	90/09						
CHA.3	006240	90/11	90/14	90/19				
CHECK	006271	8/57	45/24	45/52	46/24	46/52	47/24	48/28
		52/36	53/36	80/40				
CHORZ	006260	90/15	90/20	90/32	91/16	91/19		
CHRET	006257	90/01	90/13	90/21	90/31			
CNS	005743	85/07	85/16					
CNS1	005756	85/19	85/28					
CR	005354	76/21	76/40					
CRLF	006252	8/17	90/23					
CTR	005020	78/11	78/22	78/26				
CYC1	005734	85/08	85/38					
CYCTS	005731	84/23	85/05					
CYL	000152	6/34	64/33	64/48	73/27	74/18	74/20	
CYLN	005212	50/12	64/08	74/20				
CYLNC	005210	51/13	74/18					
D	005513	80/11	80/20					
D1	002200	29/09	30/02	65/40				
D10	002412	33/18						

D11	002431	33/34						
D12	002452	34/02						
D13	002467	34/18						
D14	002506	34/34						
D15	002525	35/02						
D16	002544	35/18						
D17	002563	35/34						
D18	002602	36/02						
D19	002620	36/18						
D2	002217	30/21						
D20	002637	37/02						
D21	002653	37/15						
D22	002672	37/31						
D23	002712	38/02						
D24	002731	38/18						
D25	002751	38/35						
D26	002773	39/02						
D27	003015	39/21						
D28	003037	39/40						
D29	003056	40/02						
D3	002241	31/02						
D30	003103	40/24						
D31	003130	41/02						
D32	003152	41/21						
D32A	003211	41/45	42/02					
D32B	003221	42/10	42/22					
D32C	003237	42/85	42/24					
D33	003240	43/02						
D34	003257	43/18						
D35	003277	43/35						
D36	003317	44/02						
D37	003335	44/17						
D4	002260	31/18						
D5	002277	31/34						
D6	002316	32/02						
D7	002335	32/18						
D8	002354	32/34						
D9	002373	33/02						
DAD1	003204	41/23	41/48					
DAD4	003207	41/40	41/51	42/17				
DECOC	006152	89/13	89/32	89/34	89/42			
DECOT	006161	89/20	89/25					
DECP	006167	89/21	89/26					
DECTB	006226	89/07	89/42					
DEVC	006413	7/41	93/01					
DEVCK	000227	7/41	7/45					
DEXT	006051	87/06	87/17					
DIRT	006665	4/03	95/52					
DIVD	006037	84/38	87/07					
DIVDO	006034	87/04	102/10					
DIVID	006033	87/03						
DLOOP	006042	87/10	87/15					
DUR=	006264	8/59	62/02	62/07	63/02	63/07	64/06	
DOSEX	006272	8/56	45/03	45/31	46/03	46/31	47/03	48/05
		50/07	50/34	51/08	51/35	52/30	53/07	54/13
		55/13	57/07	58/07	59/15	60/15	61/03	61/16
		61/29	64/22	64/39	80/19			
DPID	000155	6/39	77/13					

DRWRE	005542	80/09	80/46	80/49				
DSWT	005551	83/33	85/32	86/07				
DTOS8	000200	7/13	9/04	96/01				
DTYPE	000143	6/29	40/06	40/36	41/22	42/03	44/17	52/49
		53/27	54/02	55/02	57/26	58/27	59/02	60/02
		66/17	75/46					
E1	003355	44/19	45/02					
E10	003675	51/32	51/54					
E10E	003714	51/38	51/43	51/49				
E10.1	003677	51/31	51/36	51/45	51/50	51/52		
E10.2	003703	51/40	51/44					
E11	003726	52/29	52/54					
E11E	003743	52/33	52/42					
E11.1	003740	52/28	52/39	52/45	52/48			
E12	003763	53/06	53/32					
E12E	004001	53/10	53/15	53/26				
E12.1	003771	53/12	53/17					
E12.2	003772	53/05	53/13	53/23	53/26			
E13	004027	54/12	54/34					
E13E	004044	54/18	54/25					
E13.1	004041	54/06	54/22	54/29	54/32			
E14	004067	55/12	55/36					
E14E	004106	55/16	55/21	55/27				
E14.1	004075	55/18	55/24					
E14.2	004076	55/03	55/19	55/31	55/34			
E15	004123	57/05	57/35					
E15E	004140	57/10	57/19					
E15.1	004135	57/05	57/16	57/22	57/25			
E16	004164	58/06	58/36					
E16E	004202	58/10	58/15	58/20				
E16.1	004172	58/12	58/17					
E16.2	004173	58/05	58/13	58/23	58/26			
E17	004243	59/14	59/35					
E17E	004260	59/18	59/27					
E17.1	004255	59/10	59/24	59/31	59/34			
E18	004305	60/14	60/38					
E18E	004324	60/18	60/23	60/29				
E18.1	004313	60/20	60/26					
E18.2	004314	60/10	60/21	60/33	60/36			
E19	004342	60/39	61/02					
E1E	003403	45/06	45/16	45/22	45/28			
E2	003404	45/30						
E20	004355	61/15						
E21	004376	61/28						
E22	004403	62/02						
E22E	004442	62/18	62/24	62/34	62/40			
E23	004443	63/02						
E23E	004502	63/22	63/28	63/34	63/40			
E24	004513	63/43	64/01					
E24E	004601	64/51	65/06					
E24R	004575	64/25	64/31	64/38	64/42	64/47	64/53	65/02
E24X	004634	65/20	65/33	65/43				
E24Y	004653	65/37	65/48	65/55				
E25	004671	66/16	67/36					
E25L	004777	66/13	66/30	66/36	67/05	67/28	67/32	67/38
E25E1	004734	66/48	66/52					
E25E2	004763	67/15	67/17	67/20	67/24			
E25H	005004	66/25	66/32	67/04	67/43			

0109 .MAIN

E25L	003464	66/29	66/53	67/27				
E25M	003476	66/15	66/26	67/33	67/35	67/37		
E25T	003500	66/18	67/39					
E2E	003432	45/34	45/44	45/50	45/56			
E3	003433	46/02						
E3E	003461	46/06	46/16	46/22	46/28			
E4	003462	46/30						
E4E	003510	46/34	46/44	46/50	46/56			
E5	003511	47/02						
E5E	003537	47/06	47/16	47/22	47/28			
E6	003542	48/04						
E6E	003572	48/08	48/20	48/26	48/32			
E7	003575	50/04	50/24					
E7E	003612	50/18	50/19					
E7.1	003577	50/03	50/08	50/20	50/21			
E8	003622	50/31	50/52					
E8E	003640	50/37	50/42	50/47				
E8.1	003624	50/30	50/35	50/44	50/48	50/49		
E8.2	003630	50/39	50/43					
E9	003650	51/04	51/25					
E9E	003665	51/11	51/20					
E9.1	003652	51/03	51/09	51/21	51/23			
EGGS	000011	4/13	4/22					
ENTYM	007652	101/10	101/44					
EOCNT	003210	41/52	42/07	42/21				
EPRIN	006211	86/16	86/20					
ER	006427	93/06	93/13					
ERET	005020	86/10	86/17					
ERRCT	005644	83/22	83/28	84/34	85/04	85/17		
ESWIT	005643	83/21	83/27	84/26	85/05	86/08	86/15	86/27
E,DO	005537	88/22	88/32	88/38	88/44			
FIRST	005420	77/09	77/26					
FOUR	005202	62/09	63/09	63/15	74/09			
GADSK	005247	8/51	26/37	26/46	27/03	27/15	27/28	27/41
GENDA	006266	8/54	45/08	45/36	46/08	46/36	47/05	48/12
		50/11	51/12	52/34	54/17	57/11	59/19	62/26
		63/14	88/24					
GENRE	005127	70/09	70/14	70/15	70/19	70/24	70/25	70/34
		70/37	70/43	71/20	71/29	71/30	71/32	71/36
		71/37	71/40	72/21	72/30	72/31	72/33	73/20
		73/21	73/35	73/38	73/39			
GO	004533	64/21	65/17	65/26				
GRAN	004604	65/09	65/14					
HDN	005227	57/12	74/35					
HDNC	005225	59/20	74/33					
HEAD	000151	6/35	57/03	57/21	58/03	58/16	58/22	59/13
		59/29	60/13	60/24	60/31	74/33	74/35	
HEADE	000031	85/25	86/36					
HL	007652	101/42	101/43	101/50				
HL1	005103	87/41	87/47					
I33	006433	93/09	93/17	93/21				
I73	006435	93/12	93/20					
ICHAR	006427	84/41	86/34					
ICHK	000271	8/26	8/57					
ICRLF	000260	8/17	8/43					
IDOS	000272	8/27	8/58					
IGEN	000266	8/23	8/54					
IMESS	000261	8/18	8/42					

0110 .MAIN

INIT	006262	8/53	9/05	63/46				
IOMSK	000217	7/30	81/21	83/18				
IPDEC	006030	84/39	86/35					
IRAN	000265	8/22	65/09					
IROSK	000007	4/10	4/11					
IREAD	000267	8/24	8/55					
IRET	005564	4/05	81/24					
IS33	000255	7/49	8/14					
IS73	000256	7/48	8/15					
ISET	000250	8/09	14/03	14/12	20/15	20/32	21/05	21/22
		21/42	21/53	22/06	22/17	22/31	22/44	23/06
		23/19	23/31	23/44	24/06	24/19	25/19	25/29
		25/39	26/06	45/02	45/36	46/02	46/38	47/02
		48/04	50/04	50/31	51/04	51/32	52/29	53/06
		54/12	55/12	57/06	58/08	59/14	60/14	62/12
		63/12	88/18					
ITIN	000275	8/30	93/05					
ITR	005641	83/11	83/25	84/24	84/35	84/37		
ITRCN	000157	8/41	78/25	78/35	87/36	87/44		
ITRCT	005642	83/17	83/26	84/22	84/25			
ITRAT	000240	8/50	25/21	25/31	25/41	26/08	27/06	27/19
		27/31	27/44	37/11	71/31	72/32		
IWRT	000270	8/25	8/56					
I,ADS	000247	8/08	8/51					
I,DO	000234	8/21	8/59					
I,EH1	000233	8/20	8/39					
I,EHA	000253	8/12	8/36					
I,INI	000262	8/19	8/53					
I,INT	000246	8/07	8/50					
I,LD	000273	8/28	8/48					
I,LOO	000254	8/13	8/37					
I,RD0	000242	8/03	8/46					
I,RC1	000243	8/04	8/47					
I,RC2	000244	8/05	8/48					
I,RC3	000245	8/06	8/49					
I,S	000237	8/16	8/52					
I,SET	000251	8/10	8/35	84/21				
I,SSE	000241	8/02	8/45					
I,STU	000252	8/11	8/34					
I,WAI	000240	8/01	8/44					
KB1	000247	4/42	18/23	20/17	22/33	23/33	65/34	67/29
		78/12	79/07	81/46				
KB10	000057	4/33	26/31	26/41	26/52	26/20	26/45	29/21
		29/45	39/35	61/18	77/05			
KB11	000056	4/32	39/16	42/06	52/46	53/24	54/38	55/32
		68/49						
KB12	000055	4/31	9/25	20/03	21/19	22/14	23/16	24/18
		26/03	29/27	29/31	20/41	67/18		
KB13	000054	4/30	9/19	19/36	21/02	22/03	23/03	24/03
		25/04	25/36	29/03	29/07	29/17		
KB14	000053	4/29	9/13	19/26	20/31	20/41	30/12	31/11
		31/27	31/43	32/11	32/27	32/43	33/11	33/27
		33/43	34/11	34/27	34/43	35/11	35/27	35/43
		38/30	41/12	43/30	66/44	67/11	77/34	
KB2	000067	4/41	20/34	20/45	23/46	43/07	43/23	43/40
		44/07	44/22	44/25				
KB3	000066	4/40	21/07	23/28	24/08	41/13		
KB4	000065	4/39	21/24	23/21	24/21			

```
↑ 0005  .MAIN
01
02
        .MAIN

ADDR1   000301      1/04    2/02    2/13
ADDR2   000302      1/05    2/05    2/16
BGNAD   000466      1/41    1/46    2/42
COM2    000455      2/26    2/31    2/36
COM3    000463      2/34    2/38
COMM    000446      1/43    1/48    1/53    1/58    2/23    2/29
C.1MM   000300      1/03
EXT     000424      1/45    1/55    2/02
GETDA   000325      1/23
ILLEG   000310      1/12
IRDSK   000467      1/51    1/56    2/43
ND1     000303      1/06    2/14
ND2     000304      1/07    2/17
NORM    000435      1/40    1/50    2/13
NREAD   000375      1/30    2/20
NWRIT   000374      1/29    2/19
RESET   000321      1/17
RF 'JR  000464      1/42    1/47    1/52    1/57    2/38    2/40
SA→00   000404      1/35    1/40
SA401   000410      1/36    1/45
SA402   000414      1/37    1/50
SA403   000420      1/38    1/55
TABST   000465      2/23    2/41
WK      000320      1/16    1/17    1/20    1/23    1/26
XD1     000305      1/08    2/03
XD2     000306      1/09    2/06
XREAD   000377      1/32    2/09
XWRIT   000376      1/31    2/08
```

T67

DataGeneral

DIAGNOSTIC LISTING

LISTING

096-000171-09

PROGRAM

MOVING HEAD DISK CONTROL
RELIABILITY

TAPE

095-000068-09

ABSTRACT

THE MOVING HEAD DISK RELIABILITY PROGRAM IS A MAINTENANC
PROGRAM DESIGNED TO EXERCISE AND TEST 4046 DISK CONTROLL R
AND 1-4 DISK DRIVES. THE DISK DRIVES MAY BE SHARED BETWE N
TWO COMPUTERS IN WHICH CASE THIS PROGRAM MAY BE RUNNING N
EACH COMPUTER.

```
 0001 .MAIN MACRO REV 04.00              11:26:32 05/04/76
01
02
03
04
05
06
07              ;********************************************** **************
08              ;
09              ; NAME: MHDR.SR                              PART NUM ER: 094-000207
10              ;
11              ;
12              ; DESCRIPTION: MOVING HEAD DISK RELIABILITY PROG AM
13              ;
14              ;
15              ; REVISION HISTORY:
16              ;
17              ;       REV.            DATE
18              ;
19              ;       00              10/28/71
20              ;       01              01/14/72
21              ;       02              04/25/72
22              ;       03              11/20/72
23              ;       04              02/08/73
24              ;       05              12/12/73
25              ;       06              04/26/74
26              ;       07              02/14/75
27              ;       08              XX/XX/XX
28              ;       09              04/23/76
29              ;
30              ;
31              ; COPYRIGHT (C) DATA GENERAL CORPORATION, 1971,7 ,73,74,75,76
32              ; ALL RIGHTS RESERVED,
33              ;********************************************** **************
```

```
 0002 .MAIN
01
02              ;       MOVING HEAD DISK RELIABILITY PROGRAM
03
04              ;*******         AUTO-RUN AUTO-LOAD MODIFIED 4/7/ 2
05
06              ;1.     ABSTRACT
07
08              ;       THE MOVING HEAD DISK RELIABILITY PROGRAM
09              ;       IS A MAINTENANCE PROGRAM DESIGNED TO
10              ;       EXERCISE AND TEST THE 4046 DISK CONTROLL R
11              ;       AND 1-4 DISK DRIVES.  THE DISK DRIVES MA  BE
12              ;       SHARED BETWEEN TWO COMPUTERS IN WHICH CA E
13              ;       THIS PROGRAM MAY BE RUNNING IN EACH COMP TER.
14
15                      ;               NOTE
16                      ;               ****
17                      ;       ONE COMPUTER RUNNING RELIABILITY
18                      ;       TEST (SA 501), THE OTHER (SA 502
19                      ;       ONLY!!!
20
21              ;       THE CONTROL CAN BE DEVICE 33 OR 73.
22
23              ;2.     MACHINE REQUIREMENTS
24
25              ;       NOVA(EXCEPT MICRO)OR ECLIPSE FAMILY CENT AL PROCESSOR
26              ;       4K READ/WRITE MEMORY
27              ;       TELETYPE
28              ;       4046 DISK CONTROL
29              ;       4047 OR 4048 CONTROL ADAPTER
30              ;       1-4 MOVING HEAD DISK DRIVES
31
32              ;3.     OPERATING PROCEDURES
33
34              ;       A. VERIFY DRIVE (DRIVES) ARE READY ON-LI E
35              ;       B. LOAD PROGRAM USING BINARY LOADER OR D AGNOSTIC
36              ;          OPERATING SYSTEM.
37              ;       C. RESET AND START AT ONE OF THE
38              ;          ADDRESSES SHOWN BELOW IF LOADED WITH  INARY LOADER.
39              ;       D. IF LOADED BY DIAGNOSTIC OPERATING SYS EM,
40              ;          RELIABILITY(BELOW) WILL BE STARTED.
41
42
43              ;       STARTING ADDRESS
44              ;         2     RUN ALL
45              ;         4     CHANGE DISK CONTROL ADDRESS
46              ;       200     RELIABILITY TEST, ALL CYLINDERS( TOS START)
47              ;       501     RELIABILITY TEST, EVEN CYLINDERS
48              ;       502     RELIABILITY TEST, ODD CYLINDERS
49              ;       503     DISK ADDRESS TEST
50              ;       504     CORE DUMP ROUTINE
51              ;       505     COMMAND STRING INTERPRETER
52              ;       506     ONES TEST
53              ;       507     ZEROS TEST
54              ;       510     11011011 TEST
55              ;       511     FLOATING ONE TEST
56              ;       512     FLOATING ZERO TEST
57              ;       513     RUN ALL
58              ;       514     SEEK EXERCISER
```

```
0003 .MAIN
01
02              ;       E, 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              ;       SW1=0 FROM ERROR, DO SCOPE LOOP
09              ;       SW1=1 FROM ERROR, GO TO NEXT TEST
10              ;       SW2=1 INHIBIT TTY OUTPUT
11              ;       SW5=1 OUTPUT TO LPT
12              ;       SW6=1 HALT AFTER ERROR
13              ;       SW7=1 INHIBIT CHECK WORD AND DATA ERROR  ESSAGES
14              ;       SW8=1 BREAK TO ALLOW DISK INTERCHANGE
15              ;       SW9=1 FOR READ ONLY MODE
16
17              ;       F, STATISTICS - TYPE ANY KEY DURING
18              ;       RANDOM TESTING TO GET A REPORT OF THE
19              ;       NUMBER OF WORDS WRITTEN AND READ, PLUS
20              ;       THE NUMBER OF ERRORS.
21              ;       **** NOTE ****
22              ;       THE PROGRAM WILL ACCOUNT FOR UP TO A MAX
23              ;       OF 2*31 WORDS WRITTEN OR READ, SPECIAL
24              ;       TEST RUNS EXCEEDING THIS FACILITY WILL
25              ;       REQUIRE AN OPERATOR'S TEST LOG TO AUGMEN
26              ;       SOFTWARE ACCOUNTING.
27
28              ;3.1    PROGRAM RUNTIME
29
30              ;       PROGRAM RUNTIMES ARE SUBSTANTIALLY REDUC D WITH
31              ;       MEMORIES OF 8K OR LARGER, RUNTIMES BELOW ARE FOR
32              ;       AN 8K CORE MEMORY.
33
34              ;       RUNTIME IS DEFINED AS TIME FROM
35              ;       START TO A "PASS" MESSAGE.  TYPICAL
36              ;       RUNTIMES ARE LISTED AS FOLLOWS:
37              ;       RUNALL (SA 2): APPROX. 27 MIN.
38              ;       (NOVA 800, CORE, & 2 SURFACE DISK)
39              ;       SEEKEXERCISER (SA 514): APPROX. 5 MIN.
40              ;       (NOVA 800, CORE, & 2 SURFACE DISK)
41
42              ;3.2    BAUD RATE
43
44              ;       READ, WRITE AND SEEK OPERATIONS ARE TIME
45              ;       BY SPECIAL ROUTINES. WHEN THE PROGRAM IS
46              ;       FIRST STARTED, THE TIMING ROUTINE WILL T ST
47              ;       FOR THE PRESENCE OF A REAL TIME CLOCK (R C)
48              ;       TO DERIVE TIMING FROM IT, IF NO RTC IS
49              ;       PRESENT, THE PROGRAM WILL TYPE "TTO BAUD
50              ;       RATE". THIS MESSAGE REFERS TO THE BAUD R TE
51              ;       OF THE CONSOLE TERMINAL (DEVICE 10 & 11)
52              ;       TYPE IN THE BAUD RATE. IF A TYPING ERROR OCCURS
53              ;       IN THE NUMBER STRING (BEFORE THE CARRIAG  RETURN),
54              ;       SIMPLY TYPE A NON-NUMERIC CHARACTER AND
55              ;       THE REQUEST FOR THE BAUD RATE WILL BE
56              ;       REPEATED. IF THE CARRIAGE RETURN HAS BEE
57              ;       GIVEN AFTER A TYPING ERROR, RELOAD THE P OGRAM.
```

```
0004 .MAIN
01
02
03              ;4.     PROGRAM DESCRIPTION
04
05              ;       A, RELIABILITY TEST (SA 200)
06              ;       A RANDOM NUMBER GENERATOR IS USED TO SEL CT A
07              ;       DISK DRIVE,CYLINDER,HEAD,BEGINNING SECTO ,
08              ;       AND NUMBER OF CONSECUTIVE SECTORS.  RAND M
09              ;       DATA IS THEN GENERATED, WRITTEN, AND REA .
10              ;       THE SEQUENCE IS REPEATED INDEFINITELY.
11
12              ;       B, RELIABILITY TEST (SA 501)
13              ;       SAME AS A, EXCEPT THAT ONLY EVEN
14              ;       NUMBERED CYLINDERS ARE USED.  THIS
15              ;       ALLOWS A TWO COMPUTER SYSTEM TO RUN
16              ;       SIMULTANEOUS RELIABILITY TESTS.
17
18              ;       C, RELIABILITY TEST (SA 502)
19              ;       SAME AS B, ONLY FOR ODD NUMBERED
20              ;       CYLINDERS.
21
22              ;       D, DISK ADDRESS TEST (SA 503)
23              ;       RANDOM DATA IS FIRST WRITTEN THEN READ
24              ;       FROM ALL SECTORS ON EACH READY DISK.  TH S
25              ;       INSURES THAT ALL DISK PACK SURFACES ARE
26              ;       USEABLE AND THAT THE DISK PACK IS FORMAT ED
27              ;       PROPERLY.
28
```

```
10005 .MAIN
01
02
03              ;       F, COMMAND STRING INTERPRETER (SA 505)
04              ;       AS A TROUBLE SHOOTING AID THE SERVICE
05              ;       ENGINEER MAY TYPE IN HIS OWN TEST LOOP.
06              ;       AFTER STARTING AT 505, 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  D
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 PATTER
22
23              ;                       ALTERNATIVELY ENTER A ST ING OF
24              ;                       OCTAL 16 BIT WORDS TO BE
25              ;                       USED AS DATA,  THE WORDS
26              ;                       ENTERED ARE USED REPEATE LY
27              ;                       TO MAKE UP A SECTOR BLOC .
28              ;                       TYPE CARRIAGE TO USE THE
29              ;                       PREVIOUS ENTRY,
30
31              ;       III,    COMMAND STRING:
32
33              ;       OPTIONS 1,      READ HEAD,SECTOR,#SECTOR
34              ;               2,      WRITE SAME
35              ;               3,      SEEK CYLINDER
36              ;               4,      RECALIBRATE
37              ;               5,      LOOP (GO TO BEGINNING)
38              ;               6,      DELAY (N)  12.5MS INCREM NTS
39              ;               7,      TYPE CARRIAGE TO USE THE
40              ;                       PREVIOUS COMMAND STRING.
41
42              ;               NOTE THAT EITHER SPACES OR A COM A
43              ;               MAY BE USED AS AN ARGUMENT DELIM TER.
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 L NE.
48              ;               THE WORD "SAME" USED WITH READ,  R WRITE,
49              ;               WILL CAUSE THE PREVIOUS DISK
50              ;               ADDRESS PARAMETERS TO BE USED.
```

```
10006 .MAIN
01
02              ;       THE FOLLOWING EXAMPLE WOULD CAUSE UNIT
03              ;       1 TO REPEATEDLY SEEK  CYLINDER
04              ;       50, WRITE SECTORS 2 AND 3 WITH HEAD 5,
05              ;       THEN READ IT BACK AND CHECK,  DATA IS SP CIFIED
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              ;       G, ONES TEST  (DATA = ALL 1'S)
14              ;          ZEROS TEST  (DATA = ALL 0'S)
15              ;          110110 TEST  (DATA = WORDS OF 110110 0110110)
16              ;          FLOATING ONE TEST  (EACH SUCCESSIVE W RD
17              ;               CONTAINS ONE 1 BIT WHICH IS MOVE
18              ;               RIGHT ONE BIT EACH WORD)
19              ;          FLOATING ZERO TEST  (COMPLEMENT OF TH
20              ;               FLOATING 1 TEST)
21
22              ;       EACH OF THE ABOVE USE THE BASIC DISK ADD ESS
23              ;       TEST, SUBSTITUTING THE APPROPRIATE DATA,
24              ;       ANY OF THESE PATTERNS INCLUDING THE RAND M
25              ;       DATA USED FOR THE DISK ADDRESS TEST MAY
26              ;       BE RUN IN THE "READ ONLY" MODE, THIS IS  SEFUL
27              ;       FOR CHECKING THE INTERCHANGEABILITY OF D SK
28              ;       PACKS BETWEEN VARIOUS DISK DRIVES.
29
30              ;       TO GENERATE A DATA PATTERN START AT THE
31              ;       APPROPRIATE LOCATION AND SET SW8 TO 1,
32              ;       WHEN THE ENTIRE PACK HAS BEEN WRITTEN AN
33              ;       READ THE TELETYPE PRINTS "INTERCHANGE CI K"
34              ;       AND THE PROGRAM HALTS. THIS PACK MAY NOW
35              ;       BE READ FROM OTHER DISK DRIVES IN THE RE D
36              ;       ONLY MODE (SW9=1).
```

```
12007 .MAIN
01
02
03              ;5. ERROR REPORTING AND RECOVERY
04
05              ;       ALL PHASES OF THE SOFTWARE WORK THROUGH
06              ;       4 MAIN SUBROUTINES DESCRIBED BELOW.  EAC
07              ;       SUBROUTINE HAS A NORMAL RETURN (+3) AND
08              ;       AN ERROR RETURN (+1). EACH SUBROUTINE WA TS FOR DISK
09              ;       COMPLETION WITH INTERRUPT ENABLED.  A FA LURE
10              ;       TO DETECT INTERRUPT WITHIN 500MS (3 SEC  OR
11              ;       RECALIBRATE) RESULTS IN A "TIMEOUT" ERRO .
12
13              ;       RECALIBRATE - ANY UNUSUAL STATUS IS REPO TED
14              ;       IMMEDIATELY AND AN ERROR RETURN EXECUTED
15
16              ;       SEEK - SEEK ERROR STATUS INCREMENTS SEEK
17              ;       ERROR COUNTER.  ANY ERROR STATUS  RESULT
18              ;       IN STATUS PRINTOUT.
19
20              ;       WRITE - FOLLOWING "DONE" ON A WRITE, ERR RS ARE
21              ;       CHECKED IN THE SEQUENCE SHOWN BELOW.  ER OR
22              ;       RECOVERY PROCEDURE IS OUTLINED FOR EACH  ASE.
23              ;       IF THE ERROR IS NOT PRESENT THE NEXT CHE K IS MADE.
24
25              ;        1. READ/WRITE DONE STATUS - IF NONE, IN REMENT THE
26              ;        MISC ERROR COUNT, PRINT ILLEGAL STATUS  ESSAGE
27              ;        AND DO AN ERROR RETURN.
28
29              ;        2. MISC STATUS BITS - (ANY SEEK DONE, A Y SEEKING
30              ;        BIT, SEEK ERROR, END CYLINDER, OR DATA  ATE).
31              ;        IF ANY ERROR INCREMENT THE MISC ERROR C UNT,
32              ;        PRINT THE ILLEGAL STATUS, AND DO AN ERR R RETURN.
```

```
10008 .MAIN
01
02              ;       3. ADDRESS ERROR
03              ;             3.1 FIRST TIME - INCREMENT ADDRE S ERROR COUNT
04              ;             AND REPEAT THE WRITE.
05              ;             3.2 SECOND SUCCESSIVE FAILURE -  NCREMENT
06              ;             PERMANENT ADDRESS ERROR COUNT AN  DO A
07              ;             ERROR RETURN.
08
09              ;       4. ENDING MEMORY ADDRESS - INCREMENT TH  MISC ERROR
10              ;       COUNTER, PRINT THE ERROR MESSAGE, SET T E FATAL FLAG,
11              ;       AND GO TO 5.
12
13              ;       5. ENDING DISK ADDRESS - INCREMENT THE  ISC ERROR
14              ;       COUNTER, PRINT THE ERROR MESSAGE, SET T E FATAL FLAG,
15              ;       AND GO TO 6.
16
17              ;       6. NO FURTHER CHECKS
18              ;             6.1 FATAL SWITCH ON - DO A ERROR RETURN.
19              ;             6.2 OTHERWISE - DO A NORMAL RETU N.
```

```
10009 .MAIN
01
02              ;       READ - FOLLOWING "DONE" ON A READ, ERROR  ARE
03              ;       CHECKED IN THE SEQUENCE SHOWN BELOW,  ER OR
04              ;       RECOVERY PROCEDURE IS OUTLINED FOR EACH  ASE.
05              ;       IF THE ERROR IS NOT PRESENT THE NEXT CHE K IS MADE.
06              ;
07              ;        1. READ/WRITE DONE STATUS - IF NONE, IN REMENT THE
08              ;        MISC ERROR COUNT, PRINT ILLEGAL STATUS  ESSAGE
09              ;        AND DO AN ERROR RETURN.
10              ;
11              ;        2. MISC STATUS BITS - (ANY SEEK DONE, A Y SEEKING
12              ;        BIT, SEEK ERROR, END CYLINDER, OR DATA  ATE),
13              ;        IF ANY ERROR INCREMENT THE MISC ERROR C UNT,
14              ;        PRINT THE ILLEGAL STATUS, AND DO AN ERR R RETURN.
15              ;
16              ;        3. ADDRESS ERROR
17              ;               3,1 FIRST TIME - INCREMENT ADDRE S ERROR COUNT
18              ;               AND REPEAT THE READ.
19              ;               3,2 SECOND SUCCESSIVE FAILURE -  NCREMENT
20              ;               PERMANENT ADDRESS ERROR COUNT AN  DO A
21              ;               ERROR RETURN.
22              ;
23              ;        4. CHECK WORD ERROR
24              ;               4,1 FIRST TIME - INCREMENT THE C ECK WORD
25              ;               ERROR COUNT AND SET THE RETRY FL G.  PRINT
26              ;               "CHECK WORD ERROR" AND GO TO 5.
27              ;               4,2 SECOND SUCCESSIVE ERROR - IN REMENT THE
28              ;               PERMANENT CHECK WORD ERROR COUNT R AND SET
29              ;               THE FATAL FLAG.  PRINT "CHECK WO D ERROR"
30              ;               AND GO TO 5.
```

```
10010 .MAIN
01
02              ;       5. DATA ERROR
03              ;               5,1 FIRST TIME - SET THE RETRY F AG AND PRINT
04              ;               ERROR REPORT.
05              ;                       5,1,1 CHECK WORD ERROR - DECREMENT THE
06              ;                       CHECK WORD ERROR COUNTER AND INCREMENT
07              ;                       THE CHECK WORD & DATA ER OR COUNTER.
08              ;                       GO TO 6.
09              ;                       5,1,2 NO CHECK WORD ERRO  - INCREMENT
10              ;                       THE DATA ERROR COUNTER A D GO TO 6.
11              ;               5,2 SECOND SUCCESSIVE ERROR - SE  THE FATAL
12              ;               FLAG AND PRINT THE ERROR REPORT.
13              ;                       5,2,1 CHECK WORD ERROR - DECREMENT
14              ;                       THE PERMANENT CHECK WORD ERROR COUNTER
15              ;                       AND INCREMENT THE PERMAN NT CHECK WORD
16              ;                       & DATA ERROR COUNTER.  G  TO 6.
17              ;                       5,2,2 NO CHECK WORD ERRO  - INCREMENT
18              ;                       THE PERMANENT DATA ERROR COUNTER.
19              ;                       GO TO 6.
20              ;
21              ;       6. ENDING MEMORY ADDRESS - INCREMENT TH  MISC ERROR
22              ;       COUNTER, PRINT THE ERROR MESSAGE, SET T E FATAL FLAG,
23              ;       AND GO TO 7.
24              ;
25              ;       7. ENDING DISK ADDRESS - INCREMENT THE  ISC ERROR
26              ;       COUNTER, PRINT THE ERROR MESSAGE, SET T E FATAL FLAG,
27              ;       AND GO TO 8.
28              ;
29              ;       8. NO FURTHER CHECKS
30              ;               8,1 RETRY SWITCH ON - PRINT "TRY AGAIN"
31              ;               AND REPEAT THE TEST.
32              ;               8,2 FATAL SWITCH ON - DO A ERROR RETURN.
33              ;               8,3 NO SWITCHES ON - DO A NORMAL RETURN.
```

```
10011 .MAIN
01
02
03                                      ;DO NOT INSERT,LOC 0
04                                      ;CONTAINS DIRT POINTER
05       000000         .LOC 0
06 00000 007060         DIRT
07 00001 001021         INTERRUPT
08 00002 002003         JMP .+1         ;START HERE, RUN ALL
09 00003 000513         .RAL
10 00004 002371         JMP @NUCOD              ;CHANGE CNTL ADD S MANUALLY
11 00005 000022         0
12 00006 002044         JMP @ST
13 00007 000100 C100:   100
14
15       000022         .LOC 22
16
17 00022 000001 DRVS:   1               ;UNIT 0 CODE
18 00023 000002         2               ;     1
19 00024 000004         4               ;     2
20 00025 000010         10              ;     3
21
22 00026 000000 UNTINS: 0
23 00027 040000         40000
24 00030 100000         100000
25 00031 140000         140000
26 00032 040000 UNTDN:  40000
27 00033 020000         20000
28 00034 010000         10000
29 00035 004000         4000
30
31       000041         .LOC 41
32 00041 004220 FORT1:  TYPE
33 00042 002643         FXADD
34 00043 002644         FXADD+1
35 00044 000500 ST:     BEGIN
36 00045 001430 NEST:   EGGS            ;DO NOT INSERT,LOC 45
37                                      ;CONTAINS EGGS POINTER
38
```

```
10012 .MAIN
01 00046 002000 UNTSK:  2000            ;THESE 4 LOCS MUST STAY  OGETHER
02 00047 001000 C1000:  1000
03 00050 000400         400
04 00051 000200 C200:   200
05
06 00052 000004 CYLF:   .BLK 4          ;FROM CYLINDERS
07 00056 000004 CYLT:   .BLK 4          ;TO CYLINDERS
08 00062 000010 SEEKT:  .BLK 10         ;TOTAL SEEKS
09 00072 000004 SEKER:  .BLK 4          ;TOTAL SEEK ERRORS
10 00076 000010 WDSW:   .BLK 10         ;WORDS WRITTEN
11 00106 000010 WDSR:   .BLK 10         ;WORDS READ
12 00116 000004 CWER:   .BLK 4          ;CHECK WORD ERRORS
13 00122 000004 CWERP:  .BLK 4          ;PERM CHECK WORD ERRORS
14 00126 000004 ADDER:  .BLK 4          ;ADDRESS ERRORS
15 00132 000004 PADER:  .BLK 4          ;PERMANENT ADDR ERR
16 00136 000004 DATER:  .BLK 4          ;DATA ERRORS
17 00142 000004 CWDE:   .BLK 4          ;CHK WD AND DATA ERROR
18 00146 000004 CWDEP:  .BLK 4          ;PERM CHK WD AND DATA ER OR
19 00152 000004 PDER:   .BLK 4          ;PERMANENT DATA ERRORS
20 00156 000004 MISC:   .BLK 4          ;MISC ERRORS
21 00162 000004 SKTMO:  .BLK 4          ;SEEK/RECAL TIMEOUTS.
22 00166 000004 RWTMO:  .BLK 4          ;READ/WRITE TIMEOUTS.
23
24 00172 000000 HSW:    0              ;HEADER SWITCH
25 00173 000000 CSIF:   0              ;CMD STRING INTRP FLAG
26 00174 000000 UNIT:   0              ;CURRENT DISK DRIVE UNIT
27 00175 000000 RDYUNT: 0              ;AVAILABLE UNITS
28 00176 000000 .RDYU:  0
29 00177 000000 RALL:   0              ;RUN ALL MODE FLAG
30                                     ;DO NOT INSERT,LOC 200
31                                     ;IS OP SYS START
32       000200         .LOC 200
33 00200 002044 PROGB:  JMP @ST
34 00201 000000 RDO:    0              ;READ ONLY FLAG
35 00202 000000 RETRY:  0              ;RETRY TEST FLAG
36 00203 000000 FATAL:  0              ;FATAL TEST RESULT FLAG
37 00204 000000 MODE:   0              ;0=RECAL, 1=SEEK, 2=READ  3=WRITE
38 00205 000000 LUPSW:  0              ;LOOP ON TEST SWITCH
39 00206 000000 SSW:    0              ;SEEK SWITCH
40 00207 000207         .
41
42 00210 000002 LHD:    2              ;LAST HEAD+1;  2,10,OR 2 .
43 00211 000014 LS:     14             ;LAST SECTOR+1;  6 OR 12
44
```

```
10013 .MAIN
01
02          000023 C2=DRVS+1
03 00212    000003 C3:      3
04          000024 C4=DRVS+2
05          000025 C10=DRVS+3
06 00213    000017 C17:     17
07 00214    000033 C33:     33
08 00215    000037 C37:     37
09 00216    000040 C40:     40
10 00217    000060 C60:     60
11 00220    000067 C67:     67
12 00221    004352 STIME:   PTIME
13 00222    000313 C203,:   203,
14 00223    000212 C212:    212
15 00224    000215 C215:    215
16 00225    000377 C377:    377
17          000035 C4K=UNTDN+3
18 00226    007400 C7400:   7400
19          000034 C1OK=UNTDN+2
20 00227    077662 CSP:     77662
21 00230    177677 CSP1:    177677
22 00231    060033 CWRD:    60033
23 00232    077666 CSP3:    77666
24 00233    004576 ICALIB:  CALIB
25 00234    000000 DAXRTN:  0
```

```
10014 .MAIN
01
02 00235    000000 OMEGA:   0
03 00236    004555 IXLAY:   LAYUP
04 00237    000000 ISTAT:   0               ;STATUS AT INTERRUPT
05 00240    000000 SAV0:    0               ;SAVE AC0,1,2
06 00241    000000 SAV1:    0
07 00242    000000 SAV2:    0
08 00243    000000 SAV3:    0
09 00244    123456 RANDOM:  123456          ;CURRENT RANDOM NUMBER
10 00245    000000 RELRAN:  0
11 00246    060200 CNIOC:   NIOC 0
12 00247    007021 LAST:    PRGEND          ;PROGRAM END
13 00250    000000 LINCT:   0               ;BYTE POINTER
14 00251    000000 SEC:     0               ;STARTING SECTOR (POS)
15 00252    000000 SC:      0               ;SECTOR COUNT
16 00253    000000 HD:      0               ;HEAD
17 00254    007021 CA:      PRGEND          ;BEGINNING ADDRESS (VARI BLE)
18 00255    000000 TERM:    0               ;HOLDS TERMINATOR FLAG
19 00256    015432 UBP:     UBUFF*UBUFF     ;UNIT BYTE POINTER
20 00257    015616 CSBP:    CBUFF*CBUFF     ;CMD STRING BYTE POINTER
21 00260    007000 VARST:   VAR-1           ;VARIABLE DATA, SRT ADDR 1
22 00261    007020 VARED:   VAR+15.         ;     "       "   TABLE EN
23 00262    000000 VARSP:   0               ;     "       "   END MARK R
24 00263    000000 VARPT:   0               ;VAR POINTER
25 00264    007021 BUFF:    PRGEND          ;DATA BUFFER SRT ADDR (C NSTANT)
26 00265    014000 CMEND:   14000           ;TOP OF USABLE CORE
27 00266    000006 SMAX:    6               ;MAX SECTOR COUNT
28 00267    000000 EVODN:   0               ;CYLINDERS, (0)EVEN, (15 ODD
29 00270    000000 AECNT:   0
30 00271    000000 CWCNT:   0
31 00272    000000 DACNT:   0
32 00273    000000 RWRET:   0
33 00274    000000 DADAT:   0               ;DATA ADDRESS (DISK ADDR TEST)
34 00275    000000 CFLG:    0               ;END CYL FLAG
35 00276    000033 CDSK:    33
36 00277    000000 CSC:     0
37 00300    005200 FIRST:   BEGIN
38 00301    000000 UMSK:    0               ;UNIT MASK 1400 OR 400 O  0
39 00302    000400 HMSK:    400             ;HEAD MASK 7400 OR 17400 OR 400
40 00303    007400 SMSK:    7400            ;SECTOR MASK 7400 OR 340
41 00304    007400 LMSK:    7400            ;# SECTORS MASK 7400 OR  400
42 00305    000000 .SC:     0               ;# SECTOR TRANSFERRED ON CHK WD ERR
43 00306    000000 RLUP:    0               ;REPEAT TEST ADDRESS
44 00307    000000 ALLRET:  0
45 00310    000000 XXWRT:   0               ;WRITE RETRY COUNTER,
```

```
10P15 .MAIN
01
02 00311 000052 ICYLF:  CYLF
03 00312 001377 IALL:   ALL
04 00313 000624 IDLY:   DLTYM
05 00314 005425 ISAC:   SAC
06 00315 005431 ISTAC:  STAC
07 00316 004101 IPDEC:  PDEC
08 00317 004075 ITAC1:  PDCT
09 00320 004074 ITZ1:   ZOCT
10 00321 004043 IMESS:  MESS
11 00322 004206 ICRLF:  CRLF
12 00323 004653 ISRH:   SRH
13 00324 004735 IGATM:  GATM
14 00325 003616 IHSS:   HSS
15 00326 005435 IINP:   INP
16 00327 005115 IGEN:   G
17 00330 005114 ICHK:   C
18 00331 005237 IVAR:   VAR,0
19 00332 001151 ICLRB:  CLRB
20 00333 001163 ISET:   SET
21 00334 004664 ISM:    SM
22 00335 000675 ISTB0:  STB0
23 00336 000722 INIL:   INIL.
24 00337 001060 WAT:    WAIT
25 00340 005310 I.DBD:  .DBD
26 00341 003525 IRDAT:  RDATA
27 00342 003146 IWDAT:  WDATA
28 00343 002721 IGCS:   GCS
29 00344 003461 IR:     .READ
30 00345 003127 Iw:     .WRITE
31 00346 003261 IS:     .SEEK
32 00347 003023 IRC:    RECL
33 00350 001421 IPS:    PS
34 00351 005042 IHED:   HED
35 00352 003656 ICSW:   .CSW
36 00353 002261 ISCNT:  SCNT
37 00354 002574 ICSI:   CMDST
38 00355 002572 IQUST:  QUEST
39 00356 001057 IWT:    WAIT-1
40 00357 005253 IRAN:   RAN
41 00360 001436 I.GSD:  .GSD
42 00361 007762 ISU:    .SET
43 00362 000000 CYLNX:  0
44 00363 000000 NO.BS:  0
45 00364 000102 C102:   102
46 00365 000000 SVLMD:  0
47 00366 123456 .RAN:   123456          ;BASE RANDOM NUMBER
48 00367 002000 C2000:  2000
49 00370 000672 IFNUX:  Fx,4
50 00371 003672 NUCOD:  DXCMG
51 00372 000033 DEVICE: 33
52 00373 063077 HL:     HALT            ;FATAL ERROR,AC3
53 00374 000373         JMP .-1         ;CONTAINS CALL+1
54
55
```

```
10P16 .MAIN
01       101011 SKIP=MOV# 0,0,SKP
02       000020 IDX0=20
03       000021 IDX1=21
04       000033 .DSKP=33
05       006312 DOALL=JSR #IALL          ;DO FOLLOWING ROUTINE FO  ALL RDY UNITS
06       006357 RAND=JSR #IRAN           ;GENERATE RANDOM # IN AC
07       006314 SAVAC=JSR #ISAC          ;SAVE AC0,1,2
08       006315 SETAC=JSR #ISTAC         ;RESTORE AC0,1,2
09       006316 TYPDEC=JSR #IPDEC        ;TYPE (AC1) IN DECIMAL
10       006317 TYPAC1=JSR #ITAC1        ;TYPE (AC1) IN OCTAL
11       006320 TYPZ1=JSR #ITZ1          ;TYPE (AC1) IN OCTAL, SU P LEAD 0'S
12       006321 MESSAGE=JSR #IMESS       ;TYPE FOLLO=ING MESSAGE  N ASCII
13       006322 PCRLF=JSR #ICRLF         ;TYPE CR-LF
14       006326 INPUT=JSR #IINP          ;GET A LINE OF INPUT
15       006330 CHECK=JSR #ICHK          ;CHECK ALL OF DATA BUFFE
16       006327 GEN=JSR #IGEN            ;GENERATE DATA BUFFER
17       006325 GETPAR=JSR #IHSS         ;GET HEAD-SECT-# SECT FR M INPUT LINE
18       006324 GETATM=JSR #IGATM        ;GET NAME OR # FROM INPU  LINE
19       006323 SEARCH=JSR #ISRH         ;SEARCH FOLLOWING TABLE  OR MATCH ON AC1
20       006332 CLRRB=JSR #ICLRB         ;CLEAR READ BUFFER
21       006333 SETP=JSR #ISET           ;SET DISK PARAMETERS
22       006334 SMEM=JSR #ISM            ;SIZE MEMORY
23       006335 SETB=JSR #ISTB0          ;SET TIME BASE
24       006336 INITE=JSR #INIL          ;INITIALIZE ERROR COUNTE S ETC.
25       002350 PSTAT=JMP #IPS           ;PRINT STATUS
26       006344 READ=JSR #IR             ;DISK READ & CHECK DATA
27       006345 WRITE=JSR #Iw            ;GENERATE DATA AND WRITE DISK
28       006346 SEEK=JSR #IS             ;SEEK NEW CYLINDER
29       006347 RECAL=JSR #IRC           ;RECALIBRATE
30       006352 CKSW=JSR #ICSW           ;CHECK CONSOLE SWITCHES
31       006351 HEADER=JSR #IHED         ;PRINT ERROR MESSAGE HEA ER
32       006356 INT=T=JSR #IWT           ;INTERRUPT WAIT ROUTINE
33       006313 DELAY=JSR #IDLY          ;N MS DELAY IS IN LOC DE AY+1.
34       000401 NOP=401
35
36       006361 SETU=JSR #ISU            ;SET READY UNITS
```

```
10017 .MAIN
01                ;       ****************************************
02                ;       **      STARTING ADDRESSES              **
03                ;       ****************************************
04
05        000500          .LOC 500
06
07 00500 004415 BEGIN:    JSR STRT        ;RELIABILITY TEST, ALL C L.
08 00501 004414           JSR STRT        ;       "          EVEN
09 00502 004413           JSR STRT        ;       "          ODD
10 00503 004412           JSR STRT        ;DISK ADDRESS TEST
11 00504 006370           JSR #IFNQK      ;NOT USED...
12 00505 000474           JMP STRC        ;COMMAND STRING INTERPRE ER
13 00506 004407           JSR STRT        ;ONES TEST
14 00507 004406           JSR STRT        ;ZEROS TEST
15 00510 004405           JSR STRT        ;110110 TEST
16 00511 004404           JSR STRT        ;FLOATING ONES TEST
17 00512 004403           JSR STRT        ;FLOATING ZEROS TEST
18 00513 004402 .RAL:     JSR STRT        ;RUN ALL
19 00514 004401           JSR STRT        ;SEEK EXERCISER
```

```
10018 .MAIN
01
02                ;       ****************************************
03                ;       ***     INITIALIZATION                  *
04                ;       ****************************************
05
06 00515 020300 STRT:     LDA 0,FIRST     ;INITIALIZATION, ALL
07 00516 116400           SUB 0,3
08 00517 054517           STA 3,INDEX
09 00520 006322           PCRLF
10 00521 006321           MESSAGE         ;PRINT PRESENT DEV CODE
11 00522 006437           MSG77
12 00523 024372           LDA 1,DEVICE
13 00524 006320           TYPZ1
14 00525 006322           PCRLF
15 00526 006321           MESSAGE
16 00527 006446           MSG78           ;PRINT "SET SWITCHES"
17 00530 063077           HALT
18 00531 006334 RSTRT:    SMEM            ;SIZE MEMORY
19 00532 006335           SETB            ;SET TIME BASE
20 00533 004373           JSR HL          ;ERROR RTN FROM TIME INI IALIZE,
21 00534 006336           INITE           ;INITIALIZE BUFFERS, COU TERS, ETC.
22 00535 006361           SETU            ;SET READY UNITS
23 00536 126400           SUB 1,1         ;AC0=READY UNIT PATTERN
24 00537 101225           MOVZR 0,0,SNR
25 00540 000404           JMP RS3         ;UNIT 0 ONLY
26 00541 101224           MOVZR 0,0,SZR
27 00542 125140           MOVOL 1,1       ;UNIT 2 OR 3    MASK=3
28 00543 125140           MOVOL 1,1       ;UNIT 1         MASK=1
29 00544 125300 RS3:      MOVS 1,1
30 00545 044301           STA 1,UMSK      ;UNIT MASK
31 00546 062677           IORST
32 00547 006360           JSR #I,GSD      ;NO, GET IT
33 00550 006322           PCRLF
34 00551 006321           MESSAGE
35 00552 006345           MSG72           ;"TESTING UNIT "
36 00553 006312           DOALL
37 00554 000567           TUN             ;TYPE UNIT NUMBERS
38 00555 063511           SKPBZ TTO       ;WAIT FOR TTO
39 00556 000777           JMP .-1         ;TO FINISH
40 00557 060211           NIOC TTO        ;CLEAR DONE
41 00560 063517           SKPBZ LPT
42 00561 000777           JMP .-1
43 00562 060217           NIOC LPT
44 00563 020435           LDA 0,STR,2
45 00564 030452           LDA 2,INDEX
46 00565 113000           ADD 0,2
47 00566 003000           JMP @0,2        ;GO TO IT !!
```

```
10019 .MAIN
01
02 00567 054410 TUN:    STA 3,TUNRET    ;TYPE UNIT NUMBER
03 00570 020174         LDA 0,UNIT      ;FOLLOWED BY A ","
04 00571 024217         LDA 1,C60
05 00572 123000         ADD 1,0
06 00573 006041         JSR #FORT1
07 00574 020404         LDA 0,COMA
08 00575 006041         JSR #FORT1
09 00576 002401         JMP #TUNRET
10 00577 000000 TUNRET: 0
11 00600 000054 COMA:   ",
12
13                      ;COMMAND STRING INTERPRETER DOES NOT REQ IRE
14                      ;NORMAL INITIALIZATION.
15
16 00601 006322 STRC:   PCRLF
17 00602 006321         MESSAGE         ;PRINT PRESENT DEVICE CO E
18 00603 006437         MSG77
19 00604 024372         LDA 1,DEVICE
20 00605 006320         TYPZ1
21 00606 006322         PCRLF
22 00607 006321         MESSAGE
23 00610 006446         MSG78
24 00611 063077         HALT
25 00612 006334         SMEM            ;SIZE MEMORY
26 00613 006335         SETB            ;SET TIME BASE
27 00614 004373         JSR ML          ;ERROR RTN FROM TIME INI IALIZE,
28 00615 006336         INITE           ;INITIALIZE BUFFERS ETC,
29 00616 006360         JSR #I,GSD      ;SET LAST HEAD/SECTOR
30 00617 002407         JMP #,CMD       ;DISPATCH TO ROUTINE
31
32
33              ;       ************************************** *
34              ;       ***     DISPATCH TABLE TO PROGRAMS     *
35              ;       ************************************** *
36
37 00620 000620 STR,2:  .
38 00621 001517         RELALL          ;RELIABILITY TEST, ALL C L
39 00622 001515         RELEV           ;       "       EVEN
40 00623 001514         RELOD           ;       "       ODD
41 00624 002001         DATR            ;DISK ADDRESS TEST
42 00625 000672         FX,4            ;NOT USED
43 00626 002574 ,CMD:   CMDST           ;COMMAND STRING INTERPRE ER
44 00627 002005         DAT1            ;ONES TEST
45 00630 002007         DAT0            ;ZEROS TEST
46 00631 002011         DATP            ;125125 PATTERN
47 00632 002013         DATF1           ;FLOAT ONE
48 00633 002015         DATF0           ;FLOAT ZERO
49 00634 001740         RUNALL          ;RUN ALL PARTS
50 00635 001771         DAXSK           ;SEEK EXERCISER
51 00636 000000 INDEX:  0
```

```
10020 .MAIN
01
02 00637 000000 CSP2:   0
03 00640 160037 CIOMSK: 160037  ;****************************** *********
04                              ;**     INITIALIZATION                  **
05                              ;**          SUBROUTINES                **
06 00641 003655 .LST:   MIADD   ;****************************** *********
07 00642 000515 .LST1:  STRT
08
09                      ;CHANGE I/O ADDRESS TO 33 OR 73
10
11 00643 102401 FXADD:  SUB 0,0,SKP     ;MAKE IT 33
12 00644 020216         LDA 0,C40       ;MAKE IT 73
13 00645 040772         STA 0,CSP2
14 00646 030774         LDA 2,.LST1     ;POINTER IN MEMORY
15 00647 024771 FX,2:   LDA 1,CIOMSK    ;INST MASK
16 00650 021000         LDA 0,0,2       ;GET A WORD
17 00651 107400         AND 0,1
18 00652 034231         LDA 3,CWRD      ;IS IT AN I/O 33 OR 73 ?
19 00653 136404         SUB 1,3,SZR
20 00654 000426         JMP FX,3        ;NO
21 00655 024417         LDA 1,M41       ;MASK
22 00656 123400         AND 1,0         ;GET RID OF BIT 10
23 00657 024760         LDA 1,CSP2
24 00660 123000         ADD 1,0         ;MAKE ADDR 33 OR 73
25 00661 041000         STA 0,0,2       ;ADDR CHANGED
26 00662 151400 FX,3:   INC 2,2
27 00663 034756         LDA 3,.LST
28 00664 156404         SUB 2,3,SZR
29 00665 000762         JMP FX,2        ;MORE
30 00666 020214         LDA 0,C33       ;SET (CDSK) TO
31 00667 024750         LDA 1,CSP2      ;EQUAL THE DEVICE ADDR
32 00670 107000         ADD 0,1
33 00671 044276         STA 1,CDSK
34 00672 063077 FX,4:   HALT            ;ALL DONE,RESTART THE PR GRAM
35 00673 000777         JMP .-1
36
37 00674 177737 M41:    -41
38
```

```
!0021 .MAIN
01
02
03                      /FIND COMPUTER TIME BASE
04
05 00675 054424 STB0:   STA 3,SVSTB
06 00676 052677         IORST
07 00677 020421         LDA 0,IRUP
08 00700 040001         STA 0,1
09 00701 032233         LDA 2,@ICALIB   /SKP THE TIMER IF CALIB
10 00702 151004         MOV 2,2,SZR     /HAS BEEN LOADED.
11 00703 000413         JMP STB2
12 00704 006221         JSR @STIME      /RUN PROCESSOR TIMER.
13 00705 125015         MOV# 1,1,SNR    /CKN COUNT LOADED.
14 00706 000406         JMP STB1        /ERROR IN TIMING ROUTINE
15 00707 046233         STA 1,@ICALIB   /STORE COUNT IN CALIB.
16 00710 006236         JSR @IXLAY      /INIT THE DELAY SUBR.
17 00711 125015         MOV# 1,1,SNR    /VERIFY SUBR, IS SETUP.
18 00712 000402         JMP STB1        /,WERK IS NOT LOADED W/  MS COUNT.
19 00713 000403         JMP STB2
20 00714 034405 STB1:   LDA 3,SVSTB
21 00715 001400         JMP 0,3         /ERR IN TIME INIT, RTN T  CALL+1.
22 00716 034403 STB2:   LDA 3,SVSTB     /OTHERWISE
23 00717 001401         JMP 1,3         /RETURN TO CALL+2.
24
25 00720 001021 IRUP:   INTERRUPT
26 00721 000000 SVSTB:  0
27
28                      /CLEAR ERROR COUNTERS ETC.
29
30 00722 030311 INIL.:  LDA 2,ICYLF
31 00723 024207         LDA 1,SSW+1
32 00724 102400         SUB 0,0
33 00725 041000         STA 0,0,2
34 00726 151400         INC 2,2
35 00727 132414         SUB# 1,2,SZR
36 00730 000775         JMP .-3
37 00731 040177         STA 0,RALL
38 00732 020410         LDA 0,INIL0     /LOAD # LOOPS OF
39 00733 042410         STA 0,@INIL0+1  /RELIAB. COUNTER
40 00734 020410         LDA 0,REST
41 00735 040200         STA 0,PROGE
42 00736 020403         LDA 0,.+3
43 00737 040244         STA 0,RANDOM
44 00740 001400         JMP 0,3
45 00741 123456         123456
46 00742 001750 INIL0:  1000.
47 00743 001755         CRALL
48 00744 002044 REST:   JMP @ST
```

```
!0022 .MAIN
01
02                      /FIND AVAILABLE DISK DRIVE UNITS
03
04 00745 054414 SETU.:  STA 3,.RET
05 00746 021026         LDA 0,UNTINS,2
06 00747 063033         DOC 0,.DSKP     /SELECT THE UNIT
07 00750 020047         LDA 0,C1000
08 00751 061333         DOAP 0,.DSKP    /SEEK 0
09 00752 025046         LDA 1,UNTSK,2
10 00753 060433         DIA 0,.DSKP     /WAIT FOR UNIT TO
11 00754 123405         AND 1,0,SNR     /BEGIN SEEKING
12 00755 000776         JMP .-2
13 00756 006313         DELAY           /WAIT 13 MS
14 00757 000015         13.             /TIME CONSTANT FOR DELAY SUBR.
15 00760 002401         JMP @.RET
16 00761 000000 .RET:   0
17
18
19                      /SET READY UNITS
20
21 00762 054436 .SET:   STA 3,.SRET
22 00763 020213         LDA 0,C17       /FAKE ALL UNITS AVAILABL
23 00764 040175         STA 0,RDYUNT
24 00765 006312         DOALL           /CHECK ALL TO SEE
25 00766 000745         SETU.           /WHICH ARE THERE
26 00767 006313         DELAY
27 00770 000310         200.            /WAIT 200 MILLISECS.
28 00771 060433         DIA 0,.DSKP     /READ STATUS
29 00772 052677         IORST           /RELEASE THE ADAPTER
30 00773 126400         SUB 1,1
31 00774 101300         MOVS 0,0
32 00775 101200         MOVR 0,0
33 00776 101200         MOVR 0,0
34 00777 101200         MOVR 0,0
35 01000 101200         MOVR 0,0        /A SEEK DONE INDICATES
36 01001 125100         MOVL 1,1        /A READY UNIT
37 01002 101200         MOVR 0,0
38 01003 125100         MOVL 1,1
39 01004 101200         MOVR 0,0
40 01005 125100         MOVL 1,1
41 01006 101200         MOVR 0,0
42 01007 125100         MOVL 1,1
43 01010 044175         STA 1,RDYUNT
44 01011 125004         MOV 1,1,SZR
45 01012 002406         JMP @.SRET      /DONE
46 01013 006322         PCRLF           /NONE READY
47 01014 006321         MESSAGE
48 01015 005711         MSG16           /"NO READY UNITS"
49 01016 063077         HALT
50 01017 000744         JMP .SET+1
51 01020 000000 .SRET:  0
```

```
;0023 ,MAIN
01
02                ;       ****************************************** *****
03                ;       ***     INTERRUPT SERVICING                  **
04                ;       ****************************************** *****
05
06 01021 054243 INTER:  STA 3,SAV3
07 01022 006314         SAVAC
08 01023 065477         INTA 1          ;WHO DID IT???
09 01024 044424         STA 1,S0
10 01025 030372         LDA 2,DEVICE
11 01026 132415         SUB# 1,2,SNR
12 01027 000422         JMP INT,1       ;DISK PACK INTERRUPT
13 01030 030025         LDA 2,C10
14 01031 132415         SUB# 1,2,SNR
15 01032 000445         JMP ITTI        ;TTI INTERRUPT
16 01033 034246         LDA 3,CNIOC     ;SOMEONE ELSE
17 01034 137000         ADD 1,3
18 01035 054401         STA 3,.+1
19 01036 060200         NIOC 0          ;DEV ADDR ADDED DYNAMICA LY
20 01037 006322         PCRLF
21 01040 006321         MESSAGE
22 01041 005653         MSG9            ;INTERRUPT FROM DEVICE
23 01042 024406         LDA 1,S0
24 01043 006320         TYPZ1
25 01044 006315         SETAC
26 01045 034243         LDA 3,SAV3
27 01046 060177         INTEN
28 01047 002000         JMP @0          ;RETURN
29 01050 000000 S0:     0
30
```

```
;0024 ,MAIN
01
02
03                      ;DISK PACK INTERRUPT
04
05
06 01051 060633 INT,1:  DIAC 0,,DSKP
07 01052 040237         STA 2, ISTAT    ;SAVE FOR ALL TO USE
08 01053 030174         LDA 2, UNIT
09 01054 010421         ISZ WTRET
10 01055 002420         JMP @WTRET
11
12 01056 002000 MSK85:  2000
13
14                      ;WAIT FOR INTERRUPT
15                      ;RUN TIMER TO PREVENT HANGUP
16
17 01057 020417         LDA 0,WAYXX     ;WAIT-1 IS NORMAL ENTR T  WAIT SUBR,
18 01060 040405 WAIT:   STA 0,WAIT1     ;WAIT TIME IS IN AC0 BEF RE JSR,
19 01061 126400         SUB 1,1
20 01062 054413         STA 3,WTRET
21 01063 060177         INTEN           ;ENABLE INTERRUPTS
22 01064 006313         DELAY
23 01065 000001 WAIT1:  1
24
25 01066 060277         INTDS           ;TIMEOUT, DISABLE INTER,
26 01067 062677         IORST           ;CLEAR ADAPTER FOR 2ND C MP,
27 01070 006351         HEADER          ;PRINT
28 01071 006322         PCRLF           ;ERROR MESSAGE
29 01072 006321         MESSAGE
30 01073 006005         MSG59           ;"INTERRUPT TIMEOUT"
31 01074 002421         JMP @WTRET      ;ERROR RETURN (+1)
32 01075 000000 WTRET:  0
33 01076 003720 WAYXX:  2000,
```

```
↓0025 .MAIN
01
02                          ;TELETYPE INTERRUPT
03
04 01077 060210 ITTI:   NIOC TTI            ;KNOCK DOWN THE TTI FLAG
05 01100 020173          LDA 0,CSIF          ;CSI FLAG (SA 505)
06 01101 101004          MOV 0,0,SZR
07 01102 002354          JMP @ICSI           ;RETURN TO CSI
08 01103 020267          LDA 0,EVODN         ;IF TWO COMPUTERS
09 01104 101005          MOV 0,0,SNR         ;ONLY PRINT DURING
10 01105 000404          JMP .+4             ;READ OR WRITE
11 01106 020274          LDA 0,MODE
12 01107 101225          MOVZR 0,0,SNR
13 01110 000433          JMP ITT.5           ;A SEEK OR RECAL
14 01111 006322          PCRLF
15 01112 006322          PCRLF
16 01113 020174          LDA 0,UNIT
17 01114 040433          STA 0,SAVU
18 01115 020307          LDA 0,ALLRET        ;SAVE ALL RETURN
19 01116 040432          STA 0,SAVA
20 01117 020544          LDA 0,POT-1
21 01120 040021          STA 0,IDX1
22 01121 022021 ITT.1:   LDA 0,@IDX1         ;PRINT TABLE
23 01122 101005          MOV 0,0,SNR
24 01123 000414          JMP ITT.4           ;END TABLE, DONE.
25 01124 040405          STA 0,ITT.2
26 01125 022021          LDA 0,@IDX1
27 01126 040405          STA 0,ITT.3
28 01127 006322          PCRLF
29 01130 006321          MESSAGE
30 01131 000000 ITT.2:   0                   ;TITLE
31 01132 006312          DOALL               ;PRINT FOR ALL RDY UNITS
32 01133 000000 ITT.3:   0                   ;PRINT ROUTINE ADDRESS
33 01134 063610          SKPDN TTI           ;QUIT IF KEY STRUCK
34 01135 000764          JMP ITT.1
35 01136 060210          NIOC TTI
36 01137 020410 ITT.4:   LDA 0,SAVU          ;RESTORE UNIT #
37 01140 040174          STA 0,UNIT
38 01141 020407          LDA 0,SAVA          ;RESTORE ALL RETURN
39 01142 040307          STA 0,ALLRET
40 01143 006315 ITT.5:   SETAC               ;RESTORE AC'S
41 01144 034243          LDA 3,SAV3
42 01145 060177          INTEN
43 01146 002000          JMP @0
44 01147 000000 SAVU:    0
45 01150 000000 SAVA:    0
46
47                          ;CLEAR READ BUFFER
48
49 01151 102400 CLRB:    SUB 0,0             ;CLEAR READ BUFFER
50 01152 030264          LDA 2,BUFF          ;SPACE TO ZEROS
51 01153 024252          LDA 1,SC
52 01154 125000          MOVS 1,1
53 01155 147000          ADD 2,1
54 01156 041000          STA 0,0,2
55 01157 151400          INC 2,2
56 01160 132414          SUBM 1,2,SZR
57 01161 000775          JMP .-3
58 01162 001400          JMP 0,3
```

```
↓0026 .MAIN
01
02                          ;SET PARAMETERS
03
04 01163 054465 SET:     STA 3,SERET
05 01164 030174          LDA 2,UNIT          ;SETUP DISK CONTROL
06 01165 020252          LDA 0,SC            ;FOR CURRENT INSTRUCTION
07 01166 040305          STA 0,.SC
08 01167 034266          LDA 3,SMAX          ;MAX # SECT DUE TO MEM S ZE
09 01170 116433          SUBZM 0,3,SNC
10 01171 000460          JMP S,1             ;MEMORY TOO SMALL
11 01172 100400          NEG 0,0             ;CA-HEAD-SECTOR
12 01173 024213          LDA 1,C17           ;SECTOR COUNT
13 01174 107400          AND 0,1
14 01175 021026          LDA 0,UNTINS,2
15 01176 107000          ADD 0,1
16 01177 020251          LDA 0,SEC           ;FIRST SECTOR
17 01200 103120          ADDZL 0,0
18 01201 103120          ADDZL 0,0
19 01202 107000          ADD 0,1
20 01203 020253          LDA 0,HD
21 01204 101300          MOVS 0,0
22 01205 107000          ADD 0,1
23 01206 057233          DOCC 1,.DSKP        ;LOAD UNIT-HEAD-SECT-CNT
24 01207 062433          DIC 0,.DSKP         ;READ IT BACK
25 01210 122415          SUBM 1,0,SNR        ;CHECK
26 01211 000406          JMP SET1
27 01212 006314          SAVAC               ;LOAD ERROR
28 01213 006322          PCRLF
29 01214 006321          MESSAGE
30 01215 006152          MSG53               ;"DOC"
31 01216 000412          JMP SET2
32
33 01217 024254 SET1:    LDA 1,CA
34 01220 056033          DOB 1,.DSKP         ;LOAD CURRENT ADDRESS
35 01221 061433          DIB 0,.DSKP         ;READ IT BACK
36 01222 106415          SUBM 0,1,SNR        ;CHECK
37 01223 002425          JMP @SERET
38 01224 006314          SAVAC               ;LOAD ERROR
39 01225 006322          PCRLF
40 01226 006321          MESSAGE
41 01227 006155          MSG54               ;"DOB"
42 01230 006321 SET2:    MESSAGE
43 01231 006160          MSG55               ;"LOAD ERROR"
44 01232 006322          PCRLF
45 01233 006321          MESSAGE
46 01234 006144          MSG51               ;GOOD
47 01235 024241          LDA 1,SAV1
48 01236 006317          TYPAC1
49 01237 006321          MESSAGE
50 01240 006147          MSG52               ;BAD
```

```
10027 .MAIN
01
02 01241 024240          LDA 1,SAV0
03 01242 006317          TYPAC1
04 01243 030174          LDA 2,UNIT        ;COUNT AS A MISC ERROR
05 01244 011156          ISZ MISC,2
06 01245 006352          CKSW              ;CHECK SWITCHES
07 01246 000716          JMP SET+1
08 01247 002421          JMP @SERET
09 01250 000000 SERET:   0
10
11 01251 006314 S.1:     SAVAC
12 01252 006322          PCRLF
13 01253 006321          MESSAGE
14 01254 006102          MSG39             ;"MEM TOO SMALL
15 01255 024240          LDA 1,SAV0
16 01256 006320          TYPZ1
17 01257 006321          MESSAGE
18 01260 006115          MSG40             ;SECTORS, RESTART
19 01261 063077          HALT
20 01262 000777          JMP .-1
21
22                       ;PRINT TABLE
23
24 01263 001263          .
25 01264 005667 POT:     MSG13
26 01265 001325          PHDR
27 01266 005702          MSG15
28 01267 001335          PTSEK
29 01270 005673          MSG14
30 01271 001331          PSKER
31 01272 005775          MSG25
32 01273 001340          PTWW
33 01274 006005          MSG26
34 01275 001345          PTWR
35 01276 006042          MSG29
36 01277 001354          PTAER
37 01300 006052          MSG32
38 01301 001356          PTPAE
39 01302 006014          MSG27
40 01303 001350          PTCWE
41 01304 005026          MSG28
42 01305 001352          PTCWP
43 01306 006064          MSG31
44 01307 001360          PTDER
45 01310 006211          MSG61
46 01311 001362          PTPDE
47 01312 006247          MSG65
48 01313 001366          PTCWD
49 01314 006263          MSG66
50 01315 001370          PTPCW
51 01316 006222          MSG62
52 01317 001364          PTMS
53 01320 006373          MSG72
54 01321 001372          PSKTO
55 01322 006406          MSG73
56 01323 001374          PRWTO
57 01324 000000          0
```

```
10028 .MAIN
01 01325 054451 PHDR:    STA 3,TRET        ;PRINT HEADER NUMBER
02 01326 024174          LDA 1,UNIT
03 01327 006320          TYPZ1
04 01330 002446          JMP @TRET
05
06 01331 025072 PSKER:   LDA 1,SEKER,2     ;PRINT # SEEK ERRORS
07 01332 054444 PS.1:    STA 3,TRET
08 01333 006316          TYPDEC
09 01334 002442          JMP @TRET
10
11 01335 025062 PTSEK:   LDA 1,SEEKT,2     ;PRINT # SEEKS
12 01336 031066          LDA 2,SEEKT+4,2
13 01337 000403          JMP PT.1
14
15 01340 025076 PTWW:    LDA 1,WDSW,2      ;PRINT WORDS WRITTEN
16 01341 031102          LDA 2,WDSW+4,2
17 01342 054434 PT.1:    STA 3,TRET
18 01343 006340          JSR @I.DBD
19 01344 002432          JMP @TRET
20
21 01345 025106 PTWR:    LDA 1,WDSR,2      ;PRINT WORDS READ
22 01346 031112          LDA 2,WDSR+4,2
23 01347 000773          JMP PT.1
24
25 01350 025116 PTCWE:   LDA 1,CWER,2      ;PRINT CHECK WORD ERRORS
26 01351 000761          JMP PS.1
27
28 01352 025122 PTCWP:   LDA 1,CWERP,2     ;PRINT PERM CHK WD ERRS
29 01353 000757          JMP PS.1
30
31 01354 025126 PTAER:   LDA 1,ADDER,2     ;PRINT ADDRESS ERRORS
32 01355 000755          JMP PS.1
33
34 01356 025132 PTPAE:   LDA 1,PADER,2     ;PRINT PERM ADDR ERRS
35 01357 000753          JMP PS.1
36
37 01360 025136 PTDER:   LDA 1,DATER,2     ;PRINT DATA ERRORS
38 01361 000751          JMP PS.1
39
40 01362 025152 PTPDE:   LDA 1,PDER,2      ;PRINT PERM DATA ERRS
41 01363 000747          JMP PS.1
42
43 01364 025156 PTMS:    LDA 1,MISC,2      ;PRINT # MISC ERRORS
44 01365 000745          JMP PS.1
45
46 01366 025142 PTCWD:   LDA 1,CWDE,2      ;PRINT # CHK WD AND
47 01367 000743          JMP PS.1          ;DATA ERRORS,
48
49 01370 025146 PTPCW:   LDA 1,CWDEP,2     ;PRINT # PERM CHECK WORD
50 01371 000741          JMP PS.1          ;AND DATA ERRORS
51
52 01372 025162 PSKTO:   LDA 1,SKTMO,2     ;PRINT # OF SEEK AND
53 01373 000737          JMP PS.1          ;RECAL ERRORS,
54
55 01374 025166 PRWTO:   LDA 1,RWTMO,2     ;PRINT # OF READ/WRITE
56 01375 000735          JMP PS.1          ;TIMEOUTS,
57 01376 000000 TRET:    0
```

```
10029 .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 01377 054307 ALL:    STA 3,ALLRET
07 01400 102400         SUB 0,0
08 01401 040174 ALL.1:  STA 0,UNIT
09 01402 034307         LDA 3,ALLRET
10 01403 030174         LDA 2,UNIT
11 01404 021022         LDA 0,DRVS,2
12 01405 024175         LDA 1,RDYUNT
13 01406 123414         AND# 1,0,SZR
14 01407 007400         JSR @0,3        ;DO IT FOR THIS UNIT
15 01410 000401         JMP .+1         ;IGNORE RETURN+1,+2 ROUT S
16 01411 000401         JMP .+1
17 01412 010174         ISZ UNIT
18 01413 020174         LDA 0,UNIT
19 01414 024024         LDA 1,C4
20 01415 122414         SUB# 1,0,SZR
21 01416 000763         JMP ALL.1
22 01417 010307         ISZ ALLRET
23 01420 002307         JMP @ALLRET
24
25
26
27
28
29                      ;PRINT STATUS
30
31 01421 006351 PS:     HEADER
32 01422 006322         PCRLF
33 01423 024237         LDA 1,ISTAT
34 01424 006317         TYPAC1
35 01425 006321         MESSAGE
36 01426 005635         MSG1            ;" = ENDING STATUS"
37 01427 002273         JMP @WWRET
38
39
40 01430 000000 EGGS:   0               ;MEN FLAG
41 01431 000000         0               ;DEVICE CODE THIS RUN
42 01432 000000         0               ;NOT USED
43 01433 000000         0               ;# OF PASSES THIS RUN
44 01434 000000         0               ;RETURN ADDRESS
45 01435 000000         0               ;SWITCH REGISTER
```

```
10030 .MAIN
01
02                      ;ASK FOR DISK SIZE
03
04 01436 054425 .GSD:   STA 3,FDSRET
05 01437 006322         PCRLF
06 01440 006321         MESSAGE
07 01441 006324         MSG69           ;"TYPE THE NUMBER OF DIS  SURFACES "
08 01442 006326         INPUT           ;WAIT FOR INPUT
09 01443 000256         UBP             ;BYTE POINTER
10 01444 000445         JMP .GSD1       ;"CR" ONLY IS ERROR
11 01445 006324         GETATM          ;GET THE NUMBER
12 01446 105000         MOV 0,1
13 01447 006323         SEARCH          ;IS NUMBER 2,4,10 OR 20
14 01450 001464         HNUM            ;TABLE ADDRESS
15 01451 000440         JMP .GSD1       ;NO !!
16 01452 021011 FDS.2:  LDA 0,SMX-HNUM,2
17 01453 040211         STA 0,LS        ;LAST SECTOR+1
18 01454 021005         LDA 0,HMX-HNUM,2
19 01455 040210         STA 0,LHD       ;LAST HEAD+1
20 01456 021015         LDA 0,MS1-HNUM,2;HEAD MASK
21 01457 040302         STA 0,HMSK
22 01460 021021         LDA 0,MS2-HNUM,2;SECTOR MASK
23 01461 040303         STA 0,SMSK
24 01462 002401         JMP @FDSRET
25 01463 000000 FDSRET: 0
```

```
10031 .MAIN
01
02 01464 000002 HNUM:   2               ;CARTRIDGE
03 01465 000010         10              ;2311
04 01466 000020         20              ;2314
05 01467 000004         4               ;
06 01470 000000         0
07
08 01471 000002 HMX:    2               ;CARTRIDGE
09 01472 000012         10.             ;2311
10 01473 000024         20.             ;2314
11 01474 000004         4               ;
12
13 01475 000014 SMX:    12.
14 01476 000006         6
15 01477 000014         12.
16 01500 000014         12.
17
18 01501 000400 MS1:    400             ;HEAD MASKS
19 01502 007400         7400
20 01503 017400         17400
21 01504 001400         1400
22
23 01505 007400 MS2:    7400            ;SECTOR MASKS
24 01506 003400         3400
25 01507 007400         7400
26 01510 007400         7400
27
28 01511 006321 .GSD1:  MESSAGE         ;ERROR
29 01512 006124         MSG43           ; "-?-"
30 01513 000724         JMP .GSD+1
```

```
10032 .MAIN
01
02                ;     ************************************** ******
03                ;     ***     DISK RELIABILITY TEST          ******
04                ;     ************************************** ******
05
06 01514 102621 RELOD:  SUBZR 0,0,SKP   ;USE ODD CYLINDERS
07 01515 102520 RELEV:  SUBZL 0,0       ;USE EVEN CYLINDERS
08 01516 101001         MOV 0,0,SKP
09 01517 102400 RELALL: SUB 0,0         ;USE ALL CYLINDERS
10 01520 040267         STA 0,EVODN
11 01521 020357         LDA 0,IRAN      ;SET FOR RANDOM DATA
12 01522 042341         STA 0,@IRDAT
13 01523 042342         STA 0,@IWDAT
14 01524 020264         LDA 0,BUFF      ;MEM ADDR, CONST
15 01525 040254         STA 0,CA
16 01526 020222         LDA 0,C203,
17 01527 040506         STA 0,RATIO     ;RATIO OF REL LOOPS TO E CH SEEK TEST
18
19
20 01530 006357 REL,1:  RAND            ;GET RANDOM UNIT #
21 01531 030301         LDA 2,UMSK      ;UNIT SIZE MASK
22 01532 113700         ANDS 0,2
23 01533 021022         LDA 0,DRVS,2
24 01534 024175         LDA 1,RDYUNT    ;UNIT READY ?
25 01535 123405         AND 1,0,SNR
26 01536 000772         JMP REL,1       ;NO, TRY AGAIN
27 01537 050174         STA 2,UNIT
28
29 01540 006357 REL,2:  RAND            ;GET RANDOM HEAD #
30 01541 024302         LDA 1,HMSK      ;HEAD SIZE MASK
31 01542 123700         ANDS 1,0
32 01543 024210         LDA 1,LHD       ;HEAD # OK ?
33 01544 106437         SUBZ# 0,1,SBN
34 01545 000773         JMP REL,2       ;TOO LARGE, TRY AGAIN
35 01546 040253         STA 0,HD
36
37 01547 006357 REL,3:  RAND            ;GET RANDOM STARTING SEC OR
38 01550 024303         LDA 1,SMSK      ;SECTOR SIZE MASK
39 01551 123700         ANDS 1,0
40 01552 024211         LDA 1,LS        ;SECT # OK ?
41 01553 106437         SUBZ# 0,1,SBN
42 01554 000764         JMP REL,2       ;TOO LARGE, TRY AGAIN
43 01555 040251         STA 0,SEC
44
45 01556 006357 REL,4:  RAND            ;GET RANDOM # OF SEC
46 01557 024304         LDA 1,LMSK      ;# SECTORS MASK
47 01560 123705         ANDS 1,0,SNR    ;DON'T ALLOW 0
48 01561 020455         LDA 0,C20       ;ZERO IS ACTUALLY 20
49 01562 024266         LDA 1,SMAX      ;ENOUGH CORE ?
50 01563 106433         SUBZ# 0,1,SNC
51 01564 000772         JMP REL,4       ;NO, TRY AGAIN
52 01565 040252         STA 0,SC
53 01566 024251         LDA 1,SEC       ;CHECK TO SEE IF
54 01567 030253         LDA 2,HD        ;TRANSFER WILL CAUSE
55 01570 006314         SAVAC           ;"END CYLINDER"
56 01571 010240         ISZ SAV0
```

```
10033 .MAIN
01
02 01572 014240 REL.5:  DSZ SAV0
03 01573 000402         JMP .+2
04 01574 000420         JMP REL.6       ;SECTOR COUNT OK!
05 01575 010241         ISZ SAV1        ;CURRENT SECT+1
06 01576 020241         LDA 0,SAV1
07 01577 024211         LDA 1,LS        ;HEAD OVERFLOW ?
08 01600 122434         SUBZ# 1,0,SZR
09 01601 000771         JMP REL.5       ;NO, COUNT ANOTHER SECT
10 01602 102400         SUB 0,0
11 01603 040241         STA 0,SAV1
12 01604 010242         ISZ SAV2        ;YES, SELECT NEXT HEAD
13 01605 020242         LDA 0,SAV2
14 01606 024210         LDA 1,LHD       ;END CYLINDER ?
15 01607 122414         SUB# 1,0,SZR
16 01610 000762         JMP REL.5       ;NO, COUNT ANOTHER SECTO
17 01611 020240         LDA 0,SAV0      ;YES, DONE IF THIS IS
18 01612 101224         MOVZR 0,0,SZR   ;THE LAST SECTOR
19 01613 000743         JMP REL.4       ; GET ANOTHER RAND #
20
21 01614 006357 REL.6:  RAND            ;GET RANDOM CYLINDER #
22 01615 024225         LDA 1,C377
23 01616 123400         AND 1,0
24 01617 024267         LDA 1,EVODN
25 01620 125005         MOV 1,1,SNR
26 01621 000404         JMP REL.7       ;USE ALL CYLINDERS
27 01622 101220         MOVZR 0,0       ;CYL # RIGHT ONE
28 01623 125100         MOVL 1,1        ;ODD BIT TO (C)
29 01624 101100         MOVL 0,0        ;CYL #, ODD OR EVEN
30 01625 024222 REL.7:  LDA 1,C203,     ;IS CYL # < 203.
31 01626 122432         SUBZ# 1,0,SZC
32 01627 000765         JMP REL.6       ;NO, TRY ANOTHER #
33 01630 030174         LDA 2,UNIT
34 01631 025056         LDA 1,CYLT,2
35 01632 045052         STA 1,CYLF,2    ;"TO"="FROM"
36 01633 041056         STA 0,CYLT,2    ;NEW "TO"
37 01634 000403         JMP .+3
38 01635 000020 RATIO:  0
39 01636 000020 C20:    20
40
41                      ;**
42                      ;**     SEEK
43                      ;**
44
45 01637 102000         ADC 0,0
46 01640 040172         STA 0,#SW
47 01641 006346         SEEK            ;SEEK NEW CYLINDER
48 01642 006352         CKSW            ;ERROR, CHECK SWITCHES
49 01643 000776         JMP .-2         ;LOOP ON IT
```

```
10034 .MAIN
01
02                      ;**
03                      ;**     WRITE DISK
04                      ;**
05
06 01644 102000         ADC 0,0
07 01645 040172         STA 0,#SW
08 01646 020244         LDA 0,RANDOM    ;SAVE BEGINNING OF
09 01647 040245         STA 0,RELRAN    ;RANDOM NUMBERS
10 01650 020245         LDA 0,RELRAN    ;RESET RANDOM NUMBERS
11 01651 040244         STA 0,RANDOM    ;FOR LOOPING
12 01652 006345         WRITE           ;GEN DATA & WRITE
13 01653 006352         CKSW            ;ERROR, CHECK SWITCHES
14 01654 000774         JMP .-4         ;LOOP ON IT
15
16 01655 020267         LDA 0,EVODN     ;IF TWO COMPUTERS, SEEK  GAIN
17 01656 101005         MOV 0,0,SNR
18 01657 000406         JMP .RED        ;ONLY ONE
19 01660 102000         ADC 0,0
20 01661 040172         STA 0,#SW
21 01662 006346         SEEK            ;SEEK SAME CYLINDER
22 01663 006352         CKSW            ;ERROR, CHECK SWITCH CON ROL
23 01664 000776         JMP .-2         ;LOOP ON IT
24
25                      ;**
26                      ;**     READ DISK
27                      ;**
28
29 01665 102000 .RED:   ADC 0,0
30 01666 040172         STA 0,#SW
31 01667 020245         LDA 0,RELRAN    ;SET RAN # GEN FOR THE
32 01670 040244         STA 0,RANDOM    ;READ CHECK
33 01671 006344         READ            ;READ AND CHECK DATA
34 01672 006352         CKSW            ;ERROR, CHECK SWITCHES
35 01673 000774         JMP .-4         ;LOOP ON IT
36
37
38                      ;CHECK ENDING FOR TERMINATION
39
40 01674 020267         LDA 0,EVODN     ;IF TWO COMPUTERS DO
41 01675 101024         MOV 0,0,SZR     ;NOT RUN SEEK EXERCISER.
42 01676 000416         JMP REL.8
43 01677 014736         DSZ RATIO       ;RATIO OF REL. LOOPS TO  EEK TEST
44 01700 000414         JMP REL.8       ;NO SEEK TEST YET.
45 01701 020222         LDA 0,C203.     ;SEEK CHECKER TIME!!
46 01702 040733         STA 0, RATIO    ;REFRESH RATIO COUNTER
47 01703 020465         LDA 0,A1+7      ;GET ADDRESS OF DATA GEN RATOR
48 01704 040274         STA 0,DADAT     ;STORE FOR WRITE & READ  UBR.
49 01705 006312         DOALL           ;DO IT FOR EACH DISK
50 01706 002161         DAX.
51 01707 020357         LDA 0,IRAN      ;RELOAD FOR CELLS REENTE ING
52 01710 042341         STA 2,@IRDAT    ;RELIABILITY.
53 01711 042342         STA 2,@IWDAT
54 01712 020264         LDA 0,BUFF
55 01713 040254         STA 0,CA
```

```
10035 .MAIN
01
02
03 01714 020177 REL.8:  LDA 0,RALL
04 01715 101005         MOV 0,0,SNR
05 01716 000402         JMP .+2          ;IN REL ODD, REL EVN + R L MODE.
06 01717 000411         JMP REL.9        ;IN RUN ALL MODE.
07 01720 014435         DSZ CRALL        ;COUNT DOWN # REL LOOPS
08 01721 000607         JMP REL.1        ;AND LOOP.
09 01722 006322         PCRLF            ;
10 01723 006321         MESSAGE          ;
11 01724 006121         MSG41            ;"PASS"
12 01725 020411         LDA 0,CD1K       ;RELOAD THE
13 01726 040427         STA 0,CRALL      ;RELIAB. LOOP COUNTER
14 01727 002410         JMP @CD1K+1      ;MORE RELIABILITY
15 01730 014425 REL.9:  DSZ CRALL        ;RUN ALL MODE
16 01731 002406         JMP @CD1K+1      ;DO 512 LOOPS ONLY
17 01732 006322         PCRLF
18 01733 006321         MESSAGE
19 01734 006121         MSG41            ;"PASS"
20 01735 000403         JMP RUNALL
21 01736 001750 CD1K:   1000,            ;THESE 2 LOCS STAY TOGET ER
22 01737 001530         REL.1
23
```

```
10036 .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 01740 020225 RUNALL: LDA 0,C377
10 01741 040414         STA 0,CRALL
11 01742 040177         STA 0,RALL
12 01743 030415         LDA 2,A1-1
13 01744 050412         STA 2,PADD
14 01745 010411 R.1:    ISZ PADD
15 01746 022410         LDA 0,@PADD
16 01747 101005         MOV 0,0,SNR
17 01750 002427         JMP @RADD
18 01751 040274         STA 0,DADAT
19 01752 006312         DOALL            ;DO DISK ADDRESS
20 01753 002025         DAT.
21 01754 000771         JMP R.1          ;TEST FOR EACH PATTERN
22 01755 000000 CRALL:  0
23 01756 000000 PADD:   0
24 01757 001517 RADD:   RELALL
25
26 01760 001760         .
27 01761 005611 A1:     ZEROS
28 01762 005612         ONES
29 01763 005615         PAT1
30 01764 005620         FL1
31 01765 005626         FLZ
32 01766 005253         RAN
33 01767 000000         0
34 01770 005633         CYLADD
35
36 01771 020777 DAXSK:  LDA 0,.-1
37 01772 040274         STA 0, DADAT
38 01773 006312         DOALL
39 01774 002161         DAX.
40 01775 006322         PCRLF
41 01776 006321         MESSAGE
42 01777 006121         MSG41
43 02000 000771         JMP DAXSK
44
45
```

```
10037 .MAIN
01 02001 020765 DATR:   LDA 2,A1+5       ;DISK ADDR TEST
02 02002 024366         LDA 1,.RAN       ;SET RANDOM #,
03 02003 044244         STA 1,RANDOM     ;BASE COUNT,
04 02004 000412         JMP DATF0+1
05
06 02005 020755 DAT1:   LDA 2,A1+1
07 02006 000410         JMP DATF0+1
08
09 02007 020752 DAT0:   LDA 2,A1
10 02010 000406         JMP DATF0+1
11
12 02011 020752 DATP:   LDA 0,A1+2
13 02012 000404         JMP DATF0+1
14
15 02013 020751 DATF1:  LDA 2,A1+3
16 02014 000402         JMP DATF0+1
17
18 02015 020750 DATF0:  LDA 2,A1+4
19 02016 040274         STA 2,DADAT
20 02017 006312 DAT:    DOALL            ;DO IT ONCE FOR EACH
21 02020 002025         DAT,             ;READY DISK
22 02021 006322         PCRLF
23 02022 006321         MESSAGE
24 02023 006121         MSG41            ;"PASS"
25 02024 000773         JMP DAT
```

```
10038 .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 02025 054525 DAT,1:  STA 3,DARET
11 02026 074477         READS 3          ;IF SW9=1, READ ONLY
12 02027 024007         LDA 1,C100       ;BIT 9 MASK
13 02030 020524         LDA 2,DRD        ;READ SUBR ADDRESS,
14 02031 137404         AND 1,3,SZR
15 02032 000403         JMP .+3
16 02033 126000         ADC 1,1
17 02034 020521         LDA 0,DWT
18 02035 040455         STA 0,DATRW      ;WRITE FIRST
19 02036 044515         STA 1,RFLG       ;R/W FLAG
20 02037 020247         LDA 0,LAST
21 02040 040254         STA 0,CA         ;BUFFER ADDRESS
22 02041 020274         LDA 0,DADAT
23 02042 042341         STA 0,@IRDAT     ;DEFINE DATA
24 02043 042342         STA 0,@IWDAT     ;PATTERN
25 02044 020244         LDA 0,RANDOM
26 02045 040511         STA 0,DARAN      ;SAVE RANDOM START
27
28 02046 020510 DAT,0:  LDA 0,DARAN      ;RESET RANDOM FOR
29 02047 040244         STA 0,RANDOM     ;RECYCLE
30 02050 102400         SUB 0,0          ;INITIALIZE
31 02051 042506         STA 0,@FLO1      ;FLOATING
32 02052 042506         STA 0,@FLOZ      ;PATTERNS
33 02053 102000         ADC 0,0
34 02054 040172         STA 0,HSW        ;SET HEADER SWITCH
35 02055 006347         RECAL            ;RECALIBRATE DRIVE
36 02056 006352         CKSW             ;ERROR, CHECK SWITCHES
37 02057 000776         JMP .-2          ;LOOP ON ERROR
38
39 02060 030174 DAT,11: LDA 2,UNIT       ;SEEK NEW CYLINDER
40 02061 021056         LDA 0,CYLT,2     ;TO = FROM
41 02062 041052         STA 0,CYLF,2
42 02063 011056         ISZ CYLT,2       ;TO+1 = TO
43 02064 101400         INC 0,0
44 02065 024222         LDA 1,C203,
45 02066 122405         SUB 1,0,SNR
46 02067 000437         JMP DAT,3        ;DONE, READ OR WRITE
47 02070 102000         ADC 0,0          ;SET HEADER SWITCH
48 02071 040172         STA 0,HSW
49 02072 006346         SEEK
50 02073 006352         CKSW             ;ERROR, CHECK SWITCHES
51 02074 000776         JMP .-2          ;LOOP ON ERROR
52
53 02075 102400         SUB 0,0          ;BEGIN WITH HEAD 0
54 02076 040253         STA 0,HD         ;SECTOR 0
55 02077 040251         STA 0,SEC
56 02100 040275         STA 0,CFLG
57 02101 020266         LDA 2,SMAX       ;MAX # OF SECTORS
58 02102 040252         STA 0,SC         ;DUE TO MEM SIZE
59 02103 040277         STA 0,CSC
```

```
10039 .MAIN
01
02 02104 102000 DAT.2:  ADC 0,0          ;SET THE HEADER SWITCH
03 02105 040172         STA 0,HSW
04 02106 020244         LDA 0,RANDOM     ;SAVE RAN # POSITION
05 02107 040245         STA 0,RELRAN     ;IN CASE OF SCOPE LOOP
06 02110 020245         LDA 0,RELRAN     ;RESTORE RANDOM #
07 02111 040244         STA 0,RANDOM     ;FOR SCOPE LOOP
08 02112 006345 DATRW:  WRITE            ;READ OR WRITE
09 02113 006352         CKSW             ;ERROR, CHECK SWITCHES
10 02114 000774         JMP .-4          ;LOOP ON ERROR
11 02115 020275         LDA 0,CFLG       ;END CYL NOW ?
12 02116 101004         MOV 0.0,SZR
13 02117 000741         JMP DAT.1        ;YES - SEEK AGAIN
14 02120 004541         JSR SCNT         ;ADVANCE DISK ADDRESS
15 02121 040251         STA 0,SEC        ;UPDATE SECTOR AND
16 02122 044253         STA 1,HD         ;HEAD,
17 02123 004536         JSR SCNT         ;LOOK AHEAD
18 02124 050252         STA 2,SC         ;SET NEW SECTOR COUNT
19 02125 000757         JMP DAT.2
20
21 02126 010425 DAT.3:  ISZ RFLG
22 02127 000404         JMP DAT.4        ;DONE WRITE & READ
23 02130 020424         LDA 0,DRD
24 02131 040761         STA 0,DATRW      ;SET TO READ MODE
25 02132 000714         JMP DAT.0
26 02133 060477 DAT.4:  READS 0          ;IF SW8=1 BREAK FOR DISK
27 02134 024051         LDA 1,C200       ;PACK INTERCHANGE
28 02135 107404         AND 0,1,SZR
29 02136 000407         JMP DAT.5
30 02137 024007         LDA 1,C100       ;IF SW9=1, RESET RANDOM
31 02140 107405         AND 0,1,SNR
32 02141 002411         JMP @DARET
33 02142 024366         LDA 1,.RAN
34 02143 044244         STA 1,RANDOM
35 02144 002406         JMP @DARET
36 02145 006322 DAT.5:  PCRLF
37 02146 006321         MESSAGE
38 02147 006313         MSG68            ;"INTERCHANGE DISK"
39 02150 063077         HALT
40 02151 000777         JMP .-1
41 02152 000000 DARET:  0
42 02153 000000 RFLG:   0
43 02154 006344 DRD:    READ
44 02155 006345 DWT:    WRITE
45 02156 000000 DARAN:  0
46 02157 005617 FLO1:   FL1-1
47 02160 005625 FLOZ:   FLZ-1
```

```
10040 .MAIN
01
02
03              ;WRITE & READ THE ADDRESS OF EACH CYLINDER INTO  ACM
04              ;SECTOR THEREOF TO VERIFY EACH SEEK STROKE
05              ;PERFORMED ENDS ON THE CORRECT CYLINDER
06
07 02161 054234 DAX.1:  STA 3,DAXRTN     ;SAVE RETURN ADDRESS
08 02162 020210         LDA 0,LHD
09 02163 040365         STA 0,SVLHD
10 02164 102520         SUBZL 0,0
11 02165 103000         ADD 0,0
12 02166 040210         STA 0,LHD
13 02167 102000         ADC 0,0          ;SET THE R/W FLAG
14 02170 040763         STA 0,RFLG       ;TO WRITE (ALL 1'S)
15 02171 020764         LDA 0,DWT        ;
16 02172 040445         STA 0,DAXRW      ;WRITE FIRST
17 02173 020247         LDA 0,LAST       ;
18 02174 040254         STA 0,CA         ;BUFFER ADDRESS
19 02175 020274         LDA 0,DADAT      ;DATA GENERATOR ADDRESS  S
20 02176 042341         STA 0,@IRDAT     ;PASSED TO READ
21 02177 042342         STA 0,@IWDAT     ;AND WRITE SUBROUTINES
22 02200 102000 DAX.0:  ADC 0,0          ;
23 02201 040172         STA 0,HSW        ;SET SWITCH TO PRINT HEA ER ONCE
24 02202 102440         SUBO 0,0         ;
25 02203 040362         STA 0,CYLNX      ;CLEAR CYLINDER DATA TO  ERO,
26 02204 006347         RECAL            ;RECALIBRATE THE DRIVE,
27 02205 006352         CKSW             ;ERROR, CHECK CONSOLE SW TCHES
28 02206 000776         JMP .-2          ;LOOP ON ERROR
29 02207 101011         MOV# 0,0,SKP     ;SKIP THE LOOP INCREMENT ON 1ST PASS
30
31 02210 010362 DAX.1:  ISZ CYLNX        ;INC THE CYLINDER DATA
32 02211 030174         LDA 2,UNIT       ;
33 02212 021056         LDA 0,CYLT,2     ;PASS THE "CYLINDER TO"  ATA TO
34 02213 041052         STA 0,CYLF,2     ;"CYLINDER FROM" CELL
35 02214 011056         ISZ CYLT,2       ;INC "CYLINDER TO" DATA
36 02215 101400         INC 0,0          ;ACO INC'D TO NEW SEEK C LINDER,
37 02216 024222         LDA 1,C203,      ;CHECK (C) OF CYLT FOR B ING =313
38 02217 122415         SUB# 1,0,SNR     ;
39 02220 000433         JMP DAX.3        ;LAST CYLINDER OF R/W OP R,
40 02221 102000         ADC 0,0          ;
41 02222 040172         STA 0,HSW        ;SET SWITCH TO PRINT HEA ER ONCE,
42 02223 006346         SEEK             ;
43 02224 006352         CKSW             ;ERROR, CHECK CONSOLE SW TCHES
44 02225 000776         JMP .-2          ;LOOP ON ERROR
45 02226 102400         SUB 0,0          ;NEW CYLINDER SO
46 02227 040253         STA 0,HD         ;CLEAR ALL CONTROL CELLS
47 02230 040251         STA 0,SEC        ;BEFORE READING OR WRITI G
48 02231 040275         STA 0,CFLG       ;THIS CYLINDER,
49 02232 020266         LDA 0,SMAX       ;MAX # OF SECTORS
50 02233 040252         STA 0,SC
51 02234 040277         STA 0,CSC
```

```
10041 .MAIN
01
02
03 02235 102000 DAX,2:  ADC 0,0          ;
04 02236 040172         STA 2,HSW        ;SET HEADER SWITCH
05 02237 006345 DAXRW:  WRITE
06 02240 006352         CKSW             ;ERROR, CHECK CONSOLE SW TCHES.
07 02241 000776         JMP .-2          ;LOOP ON ERROR
08 02242 020275         LDA 0,CFLG       ;TEST THE END OF CYLINDE  FLAG.
09 02243 101004         MOV 0,0,SZR
10 02244 000744         JMP DAX,1        ;FLAG IS WAVING; DO NEXT CYLINDER.
11 02245 004414         JSR SCNT         ;NO FLAG. GO SET NEXT SE TOR & HEAD
12 02246 040251         STA 0,SEC        ;AC0 (C) STARTING SECTOR
13 02247 044253         STA 1,HD         ;AC1 (C) HEAD ADDRESS.
14 02250 004411         JSR SCNT         ;GO GET SECTOR COUNT.
15 02251 050252         STA 2,SC         ;AC2 (C) SECTOR COUNT.
16 02252 000763         JMP DAX,2        ;LOOP AND READ OR WRITE.
17
18 02253 010700 DAX,3:  ISZ RFLG         ;INC RFLG TO READ OR EXI .
19 02254 000424         JMP .+4          ;FINISHED READING
20 02255 020577         LDA 0,DRD        ;SET OPERATION
21 02256 040761         STA 0,DAXRW      ;TO READ,
22 02257 000721         JMP DAX,0        ;AND LOOP.
23 02260 000433         JMP SPRSK        ;FINISHED ADDRESS WRITTE ,
24                                       ;GO TO SEEK TESTS.
```

```
10042 .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 02261 054430 SCNT:   STA 3,SCRET
11 02262 020277         LDA 0,CSC
12 02263 040427         STA 0,WORK
13 02264 020251         LDA 0,SEC        ;CURRENT BEGINNING SECTO
14 02265 024253         LDA 1,HD         ;CURRENT BEGINNING HEAD
15
16 02266 101400 SC,1:   INC 0,0          ;SECT+1
17 02267 030211         LDA 2,LS         ;CHECK FOR SECTOR OVERFL W
18 02270 112405         SUB 0,2,SNR
19 02271 000405         JMP SC,3         ;OVERFLOW
20 02272 014420 SC,2:   DSZ WORK         ;DONE ?
21 02273 000773         JMP SC,1         ;NO
22 02274 030277         LDA 2,CSC        ;YES, EXIT
23 02275 002414         JMP @SCRET
24
25 02276 141000 SC,3:   MOV 2,0          ;SECT=0
26 02277 125400         INC 1,1          ;HEAD+1
27 02300 030210         LDA 2,LHD
28 02301 132404         SUB 1,2,SZR      ;HEAD OVERFLOW ?
29 02302 000770         JMP SC,2         ;NO
30 02303 010275         ISZ CFLG         ;YES, END CYLINDER
31 02304 034406         LDA 3,WORK
32 02305 030277         LDA 2,CSC
33 02306 172400         SUB 3,2
34 02307 151420         INC 2,2
35 02310 002401         JMP @SCRET
36 02311 000000 SCRET:  0
37 02312 000000 WORK:   0
```

```
!0043 ,MAIN
01
02                ;SUPER SEEK EXERCISER PROVIDES A SEEK SCAN
03                ;SEQUENCE CONVERGING FROM THE EXTREME OUTERMOST
04                ;TRACKS OF THE DISK INTO THE ADJACENT TRACKS IN
05                ;THE CENTER, AND THEN DIVERGING AGAIN TO THE EXT EME
06                ;OUTERMOST TRACKS.  A READ IS PERFORMED AT THE
07                ;END OF EACH SEEK TO VERIFY HEADS HAVE BEEN
08                ;POSITIONED OVER THE CORRECT TRACK.
09
10 02313 020517 SPRSK:  LDA 0,SCANX      ;GET COUNT FOR # SCANS
11 02314 040505          STA 0,XTEST      ;SAVE IN SCAN COUNTER
12 02315 020222          LDA 0,C203.
13 02316 040504          STA 0,DNSEEK     ;LOAD DOWN SEEK WITH C31 .
14 02317 014503          DSZ DNSEEK       ;DECRMT MAKES IT C312.
15 02320 102440          SUB0 0,0
16 02321 040502          STA 0,UPSEEK     ;LOAD UP SEEK WITH CYLN 0 ADDRESS.
17 02322 102520          SUBZL 0,0
18 02323 040252          STA 0,SC
19 02324 020364          LDA 0,C102       ;SUBR, SUBSEE WILL SEEK  YLINDER #
20 02325 004507          JSR SUBSEE       ;67 TO DERIVE AN INIT TI E COUNT,
21 02326 020235          LDA 0,OMEGA      ;AS CONTAINED IN OMEGA,
22 02327 040475          STA 0,BIGSK      ;FOR THE BIG SEEK TIME C LL
23 02330 040475          STA 0,LITSK      ;AND LITTLE SEEK TIME CE L,
24 02331 020471          LDA 0,DNSEEK     ;ALSO INIT THE TWO SEEK
25 02332 040474          STA 0,FBIG       ;ADDRESS CELLS, "FROM" B G SEEK,
26 02333 040476          STA 0,TLIT       ;AND "TO" LITTLE SEEK #/ 12
27 02334 020364          LDA 0,C102       ;ALSO INIT TWO OTHER SEE
28 02335 040472          STA 0,TBIG       ;ADDRESS CELLS, "TO" BIG SEEK,
29 02336 040472          STA 0,FLIT       ;AND "FROM" LITTLE SEEK  /67
30
31 02337 020464 STC:    LDA 0,UPSEEK     ;START WITH UPSEEK FIRST
32 02340 004474          JSR SUBSEE       ;GO TO SEEK SUBROUTINE,
33 02341 004570          JSR TYMCK        ;CHECK THE SEEK TIME
34 02342 004506          JSR RDCYL        ;CHECK IF HEADS ARE ON P OPER CYLINDER.
```

```
!0044 ,MAIN
01
02 02343 020457 STC.1:  LDA 0,DNSEEK     ;GET DOWN SEEK ADDRESS
03 02344 004470          JSR SUBSEE       ;SEEK SUBROUTINE
04 02345 004564          JSR TYMCK        ;CHECK SEEK TIME
05 02346 004502          JSR RDCYL        ;CHECK IF HEADS ARE ON C LINDER,
06 02347 020205          LDA 0,LUPS#      ;IF WE ARE STILL LOOPING
07 02350 101014          MOV# 0,0,SZR     ;DO NOT MOVE SEEK COUNTE S
08 02351 000766          JMP STC          ;PATH IS BETWEEN UP & DN SEEK
09 02352 010451          ISZ UPSEEK       ;INCRMT UPSEEK
10 02353 020450          LDA 0,UPSEEK     ;AND CHECK (C) FOR NOT
11 02354 024222          LDA 1,C203,      ;= TO 313,
12 02355 106415          SUB# 0,1,SNR
13 02356 000404          JMP STC,2        ;FINISHED 1 SCAN SEQUENC .
14 02357 014443          DSZ DNSEEK       ;COUNT DOWN DOWN SEEK
15 02360 000401 STC,A:   401
16 02361 000756          JMP STC          ;LOOP AND CONTINUE,
```

```
;0045 .MAIN
01
02 02362 014437 STC,2:  DSZ XTEST       ;FINISHED ALL SCANS?
03 02363 000402         JMP .+2         ;NO
04 02364 000407         JMP STC,3       ;YES, GO EXERCISE BIG & ITTLE PATHS
05 02365 024222         LDA 1,C203,     ;REFRESH DOWN SEEK
06 02366 044434         STA 1,DNSEEK
07 02367 014433         DSZ DNSEEK
08 02370 102440         SUBO 0,0
09 02371 040432         STA 0,UPSEEK    ;AND UP SEEK COUNTERS.
10 02372 000745         JMP STC         ;LOOP FOR MORE SCANS.
11
12              ;THIS SECTION EXERCISES THE SEEK PATHS FOUND TO
13              ;BE BIGGEST (SEEK) TIME AND SMALLEST (SEEK) TIME
14
15 02373 020220 STC,3:  LDA 0,C67       ;RELOAD # OF SCANS.
16 02374 040425         STA 0,XTEST     ;
17 02375 020431         LDA 0,FBIG      ;GET "FROM BIG" SEEK ADD ESS
18 02376 004436         JSR SUBSEE      ;SEEK PATH STARTS FROM " ROM" ADDRESS
19 02377 004451         JSR RDCYL
20 02400 020427         LDA 0,TBIG      ;AND ENDS AT "TO" ADDRES .
21 02401 004433         JSR SUBSEE      ;THE LONGEST SEEK TIME.
22 02402 004446         JSR RDCYL
23 02403 020425         LDA 0,FLIT      ;PATH AND THE SHORTEST
24 02404 004430         JSR SUBSEE      ;SEEK TIME PATH MEASURED ARE
25 02405 004443         JSR RDCYL
26 02406 020423         LDA 0,TLIT      ;EXERCISED HERE AND VERI IED
27 02407 004425         JSR SUBSEE      ;FOR PROPER HEAD LOCATIO .
28 02410 004440         JSR RDCYL
29 02411 014410         DSZ XTEST
30 02412 000763         JMP STC,3+2     ;KEEP ON TRUCK'IN,..
31 02413 102440         SUBO 0,0
32 02414 004420         JSR SUBSEE
33 02415 004433         JSR RDCYL
34 02416 020365         LDA 0,SVLHD
35 02417 040210         STA 0,LHD
36 02420 002234         JMP @DAXRTN     ;AND RETURN.
37
38 02421 000000 XTEST:  0
39 02422 000000 DNSEEK: 0
40 02423 000000 UPSEEK: 0
41 02424 000000 BIGSK:  0
42 02425 000000 LITSK:  0
43 02426 000000 FBIG:   0
44 02427 000000 TBIG:   0
45 02430 000000 FLIT:   0
46 02431 000000 TLIT:   0
47 02432 000004 SCANX:  4
48 02433 000000 .SEX:   0
49
```

```
;0046 .MAIN
01
02              ;SPECIAL SEEK SUBROUTINE
03
04 02434 054413 SUBSEE: STA 3, .SEA
05 02435 030174         LDA 2, UNIT
06 02436 025056         LDA 1,CYLT,2
07 02437 045052         STA 1,CYLF,2
08 02440 041056         STA 2,CYLT,2
09 02441 102000         ADC 0,0         ;SET THE HEADER PRINT
10 02442 040172         STA 0,HSW       ;ONCE SWITCH
11 02443 006346         SEEK            ;A SEEK ERROR HERE
12 02444 006352         CKSW            ;IS REPRODUCEABLE IN
13 02445 000776         JMP .-2         ;A TIGHT LOOP.
14 02446 002401         JMP @.SEA
15
16 02447 000000 .SEA:   0
17
18              ;READ CYLINDER SUBROUTINE IS CALLED AFTER SEEK T
19              ;VERIFY HEADS ARE ON PROPER CYLINDER BY READING  ATA
20              ;FIELD ON CYLINDER.
21
22 02450 054410 RDCYL:  STA 3,.SEF
23 02451 030174         LDA 2,UNIT      ;GET UNIT
24 02452 025056         LDA 1,CYLT,2    ;AND CYLINDER "ADDRESS D TA",
25 02453 044362         STA 1,CYLNX     ;AND REFRESH (C) PATTERN GEN.
26 02454 006344         READ            ;FOR READ SUBROUTINE
27 02455 000401         401             ;RETURN TO CALL+1=FATAL
28 02456 004412         JSR FTERR       ;GO TO FATAL ERROR SUBR.
29 02457 002401         JMP @.SEF       ;RETURN
30
31 02460 000000 .SEF:   0
32
```

```
10047 .MAIN
01
02
03                ;COMPARE SUBROUTINE PERFORMS A LOGICAL
04                ;COMPARISON OF (C) OF ADDRESSES IN CALL+1(ARG,A)
05                ;AND CALL+2(ARG,B), RETURNS TO CALL+3 IF ARGMT B
06                ;IS > A, RETURNS TO CALL+4 IF A>B, OR A=B,
07
08 02461 027400 GTCOMP: LDA 1,@0,3       ;ARGMT A
09 02462 023401          LDA 0,@1,3       ;ARGMT B
10 02463 106415          SUB# 0,1,SNR
11 02464 001403          JMP 3,3
12 02465 122032          ADDZ# 1,0,SZC
13 02466 001402          JMP 2,3          ;B(AC0)>A(AC1)
14 02467 001403          JMP 3,3          ;A(AC1)>B(AC0)
15
16 02470 054437 FTERR:   STA 3,,SED       ;SAVE RETURN
17 02471 020205          LDA 0,LUPSW      ;CHECK IF PROGRAM IS LOO ING
18 02472 101014          MOV# 0,0,SZR     ;IF NOT SET THIS IS 1ST  RROR.
19 02473 000413          JMP .+13
20 02474 006322          JSR @ICRLF
21 02475 006321          JSR @IMESS
22 02476 006355          MSG71            ;SEEK ERROR PATH BETWEEN
23 02477 024724          LDA 1,UPSEEK     ;GO PRINT (C) OF UPSEEK
24 02500 006320          TYPZ1
25 02501 006321          MESSAGE
26 02502 005774          MSG23            ;TAB MESSAGE
27 02503 024717          LDA 1,DNSEEK     ;PRINT (C) OF DNSEEK
28 02504 006320          TYPZ1
29 02505 006322          JSR @ICRLF
30 02506 006352          CKSW
31 02507 000402          JMP .+2
32 02510 000404          JMP .+4
33 02511 006347 FTER1:   RECAL
34 02512 000402          JMP .+2
35 02513 000411          JMP FTER2
36 02514 074477          READS 3
37 02515 177112          ADDL# 3,3,SZC    ;CKN IF CONSL SW1 IS SET
38 02516 000406          JMP FTER2
39 02517 004401          JSR .+1          ;NOT SET ...
40 02520 063077          HALT             ;HALT ON 1ST LOOP.
41 02521 034734          LDA 3,.SEF-3     ;GET NOP CODE AND USE TO
42 02522 054776          STA 3,.-2        ;REPLACE HALT FOR REST O  LOOPS.
43 02523 000766          JMP FTER1        ;RECALIBRATE LOOP.
44 02524 034404 FTER2:   LDA 3,.SED+1     ;GET HALT CODE AND
45 02525 054773          STA 3,.-5        ;RESTORE LOCATION AND
46 02526 002401          JMP @.SED        ;THEN RETURN.
47
48 02527 000000 .SED:    0
49 02530 063077          HALT             ;HOLDS HALT CODE ....
```

```
10048 .MAIN
01
02                ;SUBROUTINE WHICH IS CALLED IMMEDIATELY AFTER
03                ;SEEK AND CHECKS TIME COUNT IN OMEGA FOR
04                ;> COUNT STORED IN BIGSK OR < COUNT IN LITSK.
05
06 02531 054433 TYMCK:   STA 3,,SEB
07 02532 004727          JSR GTCOMP       ;GREATER THAN COMPARE SU ROUTINE.
08 02533 000235          OMEGA            ;ARGMT A FOR SUBROUTINE
09 02534 002424          BIGSK            ;ARGMT B FOR SUBROUTINE
10 02535 000410          JMP TYM,1        ;B>A RETURN
11 02536 024235          LDA 1,OMEGA      ;A>B SO STORE
12 02537 044665          STA 1,BIGSK      ;OMEGA IN BIG SEEK.
13 02540 030174          LDA 2,UNIT
14 02541 025056          LDA 1,CYLT,2     ;AND LOAD "TO" CYLINDER  DDRESS
15 02542 044665          STA 1,TBIG       ;INTO "TO" BIG SEEK CELL
16 02543 025052          LDA 1,CYLF,2     ;AND "FROM" CYLINDER INT
17 02544 044662          STA 1,FBIG       ;"FROM" BIG SEEK CELL.
18 02545 004714 TYM,1:   JSR GTCOMP       ;GO COMPARE
19 02546 000235          OMEGA            ;OMEGA WITH
20 02547 002425          LITSK            ;CONTENTS OF LITTLE SEEK
21 02550 101011          MOV# 0,0,SKP     ;B>A RETRN, STORE OMEGA
22 02551 000412          JMP TYM,2        ;A>B RETURN, SO EXIT.
23 02552 024235          LDA 1, OMEGA
24 02553 125015          MOV# 1,1,SNR     ;DO NOT STORE A 0,
25 02554 000407          JMP TYM,2        ;FOR VALUE OF LITTLE SEE .
26 02555 044650          STA 1,LITSK      ;STORE OMEGA IN LITTLE S EK,
27 02556 030174          LDA 2, UNIT
28 02557 025056          LDA 1,CYLT,2     ;LOAD "TO" CYLINDER ADDR SS INTO
29 02560 044651          STA 1,TLIT       ;"TO" LITTLE SEEK CELL,  ND
30 02561 025052          LDA 1,CYLF,2     ;"FROM" CYLINDER ADDRESS INTO
31 02562 044646          STA 1,FLIT       ;"FROM" LITTLE SEEK CELL
32 02563 002401 TYM,2:   JMP @.SEB        ;RETURN.
33
34 02564 000000 .SEB:    0
35 02565 000000 SQIK1:   0
36 02566 054777 FMUNIT:  STA 3,SQIK1
37 02567 102440          SUBO 0,0         ;CLEAR OUT COMMAND
38 02570 042173          STA 2,CSIF       ;STRING INTERP. FLAG.
39 02571 002405          JMP CMDST+2      ;SPECIAL ENTR. FOR UNIT  .
40
```

```
!0049 .MAIN
01
02                ;       ****************************************** *********
03                ;       ******          COMMAND STRING INTERPRET R    *****
04                ;       ****************************************** *********
05
06                        ;GET COMMAND STRING PARAMETERS
07                        ;UNIT SELECT
08 02572 006321 QUEST:    MESSAGE
09 02573 006124           MSG43           ; =?=
10 02574 102520 CMDST:    SUBZL 0,0       ;SET THE CSI FLAG
11 02575 040173           STA 0,CSIF
12 02576 062210           NIOC TTI        ;CLEAR TTI FLAG
13 02577 062233           NIOC .DSKP      ;CLEAR DISK GARBAGE
14 02600 006322           PCRLF
15 02601 006321           MESSAGE
16 02602 005721           MSG17           ;"UNIT: "
17 02603 006326           INPUT           ;ACCEPT INPUT
18 02604 000256           UBP
19 02605 000422           JMP CMD,2       ;SAME AS LAST TIME
20 02606 102400           SUB 0,0
21 02607 040175           STA 0,RDYUNT    ;CLEAR ALL UNITS
22 02610 006324           GETATM
23 02611 125004           MOV 1,1,SZR     ;AC0=# AC1=NAME
24 02612 000760           JMP QUEST       ;NAMES NOT LEGAL
25 02613 024024           LDA 1,C4
26 02614 122432           SUBZ# 1,0,SZC
27 02615 000755           JMP QUEST       ;UNIT #>3
28 02616 040174           STA 0,UNIT
29 02617 101003           MOV 0,0,SNC
30 02620 000752           JMP QUEST       ;MORE LEFT, ERROR
31 02621 024173           LDA 1,CSIF
32 02622 125005           MOV 1,1,SNR
33 02623 002742           JMP @SQIK1
34 02624 000473           JMP CMD,2
```

```
!0050 .MAIN
01
02                        ;DATA SELECT
03
04 02625 006321           MESSAGE
05 02626 006124           MSG43           ; =?=
06 02627 006322 CMD,2:    PCRLF
07 02630 006321           MESSAGE
08 02631 005725           MSG18           ;"DATA: "
09 02632 006326           INPUT           ;ACCEPT INPUT
10 02633 000256           UBP
11 02634 000445           JMP GCS         ;SAME AS LAST TIME
12 02635 006324           GETATM
13 02636 125005           MOV 1,1,SNR
14 02637 000412           JMP CMD,3       ;NO NAME, MUST BE #
15 02640 006323           SEARCH          ;SEARCH NAME TABLE
16 02641 005574           DNT
17 02642 000753           JMP CMD,2-2     ;NOT FOUND
18 02643 025027           LDA 1,DNT,1-DNT,2
19 02644 046341           STA 1,#IRDAT
20 02645 046342           STA 1,#IWDAT
21 02646 101005           MOV 0,0,SNR
22 02647 000756           JMP CMD,2-2     ;NO CR FOLLOWING NAME
23 02650 000431           JMP GCS         ;GET COMMAND STRING
```

```
!0051 .MAIN
01
02 02651 024260 CMD.3:  LDA 1,VARST
03 02652 044020         STA 1,IDX0
04 02653 042020         STA 0,@IDX0
05 02654 024255         LDA 1,TERM
06 02655 125004         MOV 1,1,SZR
07 02656 000416         JMP CMD.5
08 02657 006324 CMD.4:  GETATM
09 02660 125004         MOV 1,1,SZR
10 02661 000744         JMP CMD.2-2     ;NAMES NOT ALLOWED
11 02662 042020         STA 0,@IDX0
12 02663 101002         MOV 0,0,SZC
13 02664 000410         JMP CMD.5
14 02665 020251         LDA 0,VARED
15 02666 024020         LDA 1,IDX0
16 02667 122414         SUB# 1,0,SZR
17 02670 000767         JMP CMD.4
18 02671 006322         PCRLF
19 02672 006321         MESSAGE
20 02673 005742         MSG20
21
22 02674 020020 CMD.5:  LDA 0,IDX0
23 02675 040262         STA 0,VARSP
24 02676 020331         LDA 0,IVAR
25 02677 042341         STA 0,@IRDAT
26 02700 042342         STA 0,@IWDAT
27
28
29
30                      ;GET COMMAND STRING
31
32 02701 006322 GCS:    PCRLF
33 02702 006321         MESSAGE
34 02703 005731         MSG19           ;"COMMAND STRING: "
35 02704 006326         INPUT           ;ACCEPT INPUT
36 02705 000257         CSBP            ;CMD STR BYTE POINTER
37 02706 000421         JMP .+1         ;SAME AS BEFORE
38 02707 063710 GCS.1:  SKPDZ TTI
39 02710 000664         JMP CMDST       ;INTERRUPT CMD STRING
40 02711 102000         ADC 0,0
41 02712 040172         STA 0,MSW
42 02713 006324         GETATM          ;LINCT=BYTE POINTER
43 02714 006323         SEARCH
44 02715 005557         DCT             ;NAME IN AC1
45 02716 000402         JMP CSER        ;NOT FOUND
46 02717 003007         JMP @DCT,1-DCT,2;DISPATCH
47
48 02720 006321 CSER:   MESSAGE         ;COMMAND STRING ERROR
49 02721 006124         MSG43           ; -?-
50 02722 000757         JMP GCS
```

```
!0052 .MAIN
01
02                      ;READ COMMAND
03
04 02723 101002 RE:     MOV 0,0,SZC
05 02724 000774         JMP CSER        ;CR TERMINATOR ILLEGAL
06 02725 006325         GETPAR          ;GET R/W PARAMETERS
07 02726 020245         LDA 0,RELRAN    ;SET RAN # BEGIN
08 02727 040244         STA 0,RANDOM
09 02730 006344         READ            ;READ AND CHECK DATA
10 02731 006352         CKSW            ;ERROR, LOOK AT SWITCHES
11 02732 000774         JMP .-4         ;LOOP ON ERROR
12 02733 020255 RE1:    LDA 0,TERM
13 02734 101024         MOV 0,0,SZR
14 02735 000637         JMP CMDST               ;ALL DONE
15 02736 000751         JMP GCS.1       ;GET ANOTHER COMMAND
16
17                      ;WRITE COMMAND
18 02737 101002 WT:     MOV 0,0,SZC
19 02740 000760         JMP CSER        ;CR TERMINATOR ILLEGAL
20 02741 006325         GETPAR          ;GET R/W PARAMETERS
21 02742 020244         LDA 0,RANDOM    ;SAVE BEGIN OF RANDOM
22 02743 040245         STA 0,RELRAN    ;NUMBERS IN CASE OF LOOP
23 02744 020245         LDA 0,RELRAN    ;RESET RAN # BEGINNING I
24 02745 040244         STA 0,RANDOM    ;THE SCOPE LOOP
25 02746 006345         WRITE           ;GENERATE DATA AND WRITE
26 02747 006352         CKSW            ;ERROR, CHECK SWITCHES
27 02750 000774         JMP .-4         ;LOOP ON ERROR
28 02751 000762         JMP RE1
```

```
10053 .MAIN
01
02                      ;SEEK COMMAND
03
04 02752 101022 SK:     MOV 2,2,SZC
05 02753 000745         JMP CSER        ;CR TERMINATOR ILLEGAL
06 02754 006324         GETATM          ;GET CYL # IN AC0
07 02755 152560         SUBCL 2,2
08 02756 050255         STA 2,TERM
09 02757 125004         MOV 1,1,SZR
10 02760 000740         JMP CSER        ;NAMES ILLEGAL
11 02761 030174         LDA 2,UNIT
12 02762 025056         LDA 1,CYLT,2    ;TO = CYL #
13 02763 045052         STA 1,CYLF,2    ;TO BECOMES FROM
14 02764 041056         STA 0,CYLT,2    ;SET NEW TO
15 02765 006346         SEEK            ;DO THE SEEK
16 02766 006352         CKSW            ;ERROR, LOOK AT SWITCHES
17 02767 000776         JMP .-2         ;LOOP ON ERROR
18 02770 000743         JMP RE1
19
20                      ;RECALIBRATE
21
22 02771 102560 RCL:    SUBCL 0,0
23 02772 040255         STA 0,TERM      ;SAVE (C), 1=TERM
24 02773 006347         RECAL           ;RECALIBRATE
25 02774 006352         CKSW            ;ERROR, CHECK SWITCHES
26 02775 000776         JMP .-2         ;LOOP ON ERROR
27 02776 030174         LDA 2,UNIT      ;INCRMT CYLN TO 0
28 02777 011056         ISZ CYLT,2      ;ON RETURN FROM RECAL
29 03000 000421         NOP             ;FOR CMMD STRING INTERP.
30 03001 000732         JMP RE1
31
32                      ;LOOP
33
34 03002 020257 LUP:    LDA 0,CSBP      ;RESET THE COMMAND STRIN
35 03003 040250         STA 0,LINCT     ;BYTE POINTER
36 03004 102000         ADC 0,0         ;SET THE HEADER FLAG
37 03005 040172         STA 0,HSW
38 03006 000771         JMP GCS,1
39
40                      ;DELAY
41
42 03007 101002 DLAY:   MOV 0,0,SZC
43 03010 000710         JMP CSER
44 03011 006324         GETATM          ;GET DELAY #
45 03012 125005         MOV 1,1,SNR
46 03013 101002         MOV 0,0,SZC     ;NAMES ILLEGAL
47 03014 000704         JMP CSER        ;CR TERM IS ILLEGAL
48 03015 126520         SUBZL 1,1       ;DELAY COUNT OF 0 NOT
49 03016 123000         ADD 1,0         ;PERMITTED.
50 03017 040402         STA 0,.+2
51 03020 006313         DELAY
52 03021 000000         0
53 03022 000665         JMP GCS,1
54
```

```
10054 .MAIN
01
02              ;       ****************************************** ******
03              ;       *****           DISK HANDLER SUBROUTINES  *****
04              ;       ****************************************** ******
05
06                      ;RECALIBRATE
07                      ;RETURN+1  FATAL ERROR
08                      ;RETURN+3  NORMAL
09
10 03023 054273 RECL:   STA 3,RWRET
11 03024 102400         SUB 0,0         ;SET THE PROGRAM MODE
12 03025 042204         STA 2,MODE
13 03026 030174         LDA 2,UNIT
14 03027 021026         LDA 0,UNTINS,2
15 03030 063233         DOCC 0,,DSKP    ;SELECT UNIT
16 03031 020427         LDA 0,C1400
17 03032 061333         DOAP 0,,DSKP    ;RECAL
18 03033 021055         LDA 0,CYLT,2
19 03034 041052         STA 0,CYLF,2    ;TO = FROM
20 03035 102000         ADC 0,0
21 03036 041056         STA 0,CYLT,2
22 03037 020420         LDA 0,RCTYM     ;4 SECOND TIMER
23 03040 006337         JSR @WAT        ;WAITING FOR INTERRUPT
24 03041 101011         SKIP
25 03042 000404         JMP .+4
26 03043 030174         LDA 2,UNIT      ;RECAL TIMEOUTS ARE
27 03044 011162         ISZ SKTMO,2     ;COUNTED HERE !!
28 03045 002273         JMP @RWRET      ;TIMEOUT!!!
29 03046 025032         LDA 1,UNTON,2   ;RECAL INT. RETURN
30 03047 107405         AND 0,1,SNR
31 03050 002350         PSTAT           ;NO DONE STATUS
32 03051 124220         COM 1,1
33 03052 034230         LDA 3,CSP1      ;177677
34 03053 167400         AND 3,1
35 03054 107404         AND 0,1,SZR
36 03055 002350         PSTAT           ;BAD STATUS
37 03056 000562         JMP GENRET
38 03057 007640 RCTYM:  4000.
39 03060 001400 C1400:  1400
```

```
10055 .MAIN
01
02                      ;SEEK SUBROUTINE
03                      ;RETURN+3 NORMAL
04                      ;RETURN+1 FATAL ERROR
05
06 03061 054273 .SEEK:  STA 3,RWRET
07 03062 102520         SUBZL 0,0           ;SET THE PROGRAM MODE
08 03063 040204         STA 0,MODE
09 03064 030174         LDA 2,UNIT
10 03065 021026         LDA 0,UNTINS,2
11 03066 063233         DOCC 0,.DSKP        ;SELECT UNIT
12 03067 021056         LDA 0,CYLT,2        ;CYL #
13 03070 024047         LDA 1,C1000
14 03071 123000         ADD 1,0
15 03072 061333         DOAP 0,.DSKP        ;SEEK!!
16 03073 011066         ISZ SEEKT+4,2       ;COUNT SEEKS
17 03074 000402         JMP .+2
18 03075 011062         ISZ SEEKT,2         ;DOUBLE PRES.
19 03076 000401         NOP
20 03077 006356         INTWT               ;WAIT FOR INTERRUPT
21 03100 101011         SKIP
22 03101 000404         JMP .+4
23 03102 030174         LDA 2,UNIT          ;INCRMT THE SEEK
24 03103 011162         ISZ SKTMO,2         ;TIMEOUTS COUNTER !!
25 03104 002273         JMP @RWRET          ;TIMEOUT!!!
26
27 03105 025032         LDA 1,UNTDN,2       ;INTERRUPT RETURN AC2=UN T #
28 03106 107415         ANDZ# 0,1,SNR       ;AC0 = STATUS
29 03107 000407         JMP SE,1            ;IMPROPER DONE STATUS
30 03110 124000         COM 1,1
31 03111 034230         LDA 3,CSP1          ;177677
32 03112 167420         AND 3,1
33 03113 107404         AND 0,1,SZR
34 03114 000402         JMP SE,1            ;WRONG STATUS
35 03115 000523         JMP GENRET
36
37 03116 024216 SE,1:   LDA 1,C40           ;SEEK ERR ??
38 03117 107404         AND 0,1,SZR
39 03120 011072         ISZ SEKER,2         ;COUNT SEEK ERRORS
40 03121 002350         PSTAT
41 03122 063077         HALT
42 03123 000777         JMP .-1
43 03124 000420 C400:   400
44 03125 003442 FTWRT:  FLWRT
45
```

```
10056 .MAIN
01
02                      ;WRITE SUBROUTINE
03                      ;RETURN+3, NORMAL
04                      ;RETURN+1, FATAL ERROR
05
06 03126 003147         WDATA+1             ;RETRY ADDRESS
07 03127 054273 .WRITE: STA 3,RWRET
08 03130 020776         LDA 0,.WRITE-1      ;SET THE RETRY ADDRESS
09 03131 040306         STA 0,RLUP
10 03132 020212         LDA 0,C3            ;SET THE PROGRAM MODE
11 03133 040204         STA 0,MODE
12 03134 040310         STA 0,XXWRT
13 03135 102000         ADC 0,0
14 03136 040270         STA 0,AECNT
15 03137 040271         STA 0,CWCNT
16 03140 102400         SUB 0,0
17 03141 040203         STA 0,FATAL         ;CLEAR FATAL FLAG
18 03142 040363         STA 0,NO.BS         ;CLEAR BAD SECTOR COUNTE
19 03143 020252         LDA 0,SC
20 03144 040305         STA 0,.SC           ;# SECTORS TO XFER
21 03145 006327         GEN
22 03146 000000 WDATA:  0                   ;ADDR OF DATA GEN ROUT
23 03147 006333         SETP                ;SET PARAMETERS
24 03150 030174         LDA 2,UNIT
25 03151 021056         LDA 0,CYLT,2
26 03152 024752         LDA 1,C400
27 03153 123000         ADD 1,0
28 03154 051133         DOAS 0,.DSKP        ;WRITE
29 03155 006356         INTWT               ;WAIT FOR INTERRUPT
30 03156 101011         SKIP
31 03157 000404         JMP .+4
32 03160 030174         LDA 2,UNIT          ;INCRMT THE READ/WRITE
33 03161 011166         ISZ RWTMO,2         ;TIMEOUTS COUNTER !!
34 03162 002273         JMP @RWRET          ;TIMEOUT
35
36 03163 101113         MOVL# 0,0,SNC       ;INTERRUPT RETURN
37 03164 002741         JMP @FTWRT          ;NO R/W DONE FLAG
38 03165 024232         LDA 1,CSP3          ;077666
39 03166 107404         AND 0,1,SZR
40 03167 002736         JMP @FTWRT          ;FATAL STATUS
41 03170 024025         LDA 1,C10
42 03171 107404         AND 0,1,SZR
43 03172 000472         JMP RE,2            ;ADDRESS ERROR
44 03173 051433         DIB 0,.DSKP         ;READ MEM ADDR
45 03174 024252         LDA 1,SC            ;SECTOR COUNT
46 03175 125300         MOVS 1,1            ;WORD COUNT
47 03176 034254         LDA 3,CA            ;STARTING ADDRESS
48 03177 167000         ADD 3,1
49 03200 125400         INC 1,1
50 03201 125400         INC 1,1             ;ADDR+WC SHOULD EQUAL
51 03202 106414         SUB# 0,1,SZR        ;THE ENDING ADDRESS
52 03203 000443         JMP WE,1            ;ERROR
```

```
10057 .MAIN
01
02 03204 020252 .W1:    LDA 0,SC        ;FIND ENDING DISK
03 03205 040277         STA 0,CSC       ;ADDRESS
04 03206 006353         JSR @ISCNT      ;AC0=SECT, AC1=HEAD
05 03207 101004         MOV 0,0,SZR     ;IF THE SECT = 0 THE LAS
06 03210 000404         JMP .+4         ;ICR HEAD DID NOT OCCUR
07 03211 020211         LDA 0,LS        ;SECT = LAST SECT
08 03212 124400         NEG 1,1         ;HEAD =1
09 03213 124000         COM 1,1
10 03214 030174         LDA 2,UNIT
11 03215 035026         LDA 3,UNTINS,2  ;FORM THE ENDING DISK
12 03216 125300         MOVS 1,1        ;ADDRESS AS READ BY "DIC
13 03217 167000         ADD 3,1
14 03220 103120         ADDZL 0,0
15 03221 103120         ADDZL 0,0       ; SECT*4
16 03222 107000         ADD 0,1         ;EXPECTED ADDRESS
17 03223 062433         DIC 0,,DSKP     ;READ ACTUAL
18 03224 106414         SUB# 0,1,SZR
19 03225 000427         JMP @E,2        ;ENDING DISK ADDRESS ERR R
20
21 03226 020252 .W2:    LDA 2,SC        ;ADD WORDS WRITTEN
22 03227 101300         MOVS 0,0        ;TO TOTAL COUNT
23 03230 025102         LDA 1,@DSW+4,2
24 03231 107022         ADDZ 0,1,SZC    ;DOUBLE PRECISION
25 03232 011076         ISZ @DSW,2
26 03233 000401         NOP
27 03234 045102         STA 1,@DSW+4,2
28 03235 020273         LDA 0,FATAL     ;WAS THERE A FATAL ERROR ?
29 03236 101004         MOV 0,0,SZR
30 03237 002273         JMP @RWRET      ;ERROR RETURN
31 03240 020205 GENRET: LDA 0,LUPSW     ;IS THE LOOP SWITCH ON ?
32 03241 101004         MOV 0,0,SZR
33 03242 002273         JMP @RWRET      ;ERROR RETRUN
34 03243 010273         ISZ RWRET
35 03244 010273         ISZ RWRET
36 03245 002273         JMP @RWRET      ;NORMAL RETURN
37
38 03246 004550 WE,1:   JSR SWCE        ;ENDING MEM ADDR ERROR
39 03247 006172         MSG57           ;IDENTIFIER
40 03250 030174         LDA 2,UNIT
41 03251 011156         ISZ MISC,2      ;COUNT AS MISC TYPE ERRO
42 03252 010203         ISZ FATAL       ;SET FATAL FLAG
43 03253 000731         JMP .W1         ;GO ON
44
45 03254 004542 WE,2:   JSR SWCE        ;ENDING DISK ADDR ERROR
46 03255 006277         MSG67
47 03256 030174         LDA 2,UNIT
48 03257 011156         ISZ MISC,2      ;COUNT AS MISC TYPE ERRO
49 03260 010203         ISZ FATAL       ;SET FATAL FLAG
50 03261 000745         JMP .W2         ;GO ON
```

```
10058 .MAIN
01
02 03262 011156 RE,1:   ISZ MISC,2      ;COUNT AS MISC TYPE ERRO
03 03263 002350         PSTAT           ;PRINT STATUS & ERR RETU N
04
05 03264 006351 RE,2:   HEADER          ;ADDRESS ERROR
06 03265 006322         PCRLF
07 03266 006321         MESSAGE
08 03267 006042         MSG29           ;"ADDRESS ERROR"
09 03270 030174         LDA 2,UNIT
10 03271 010270         ISZ AECNT       ;ADDRESS ERROR COUNTER
11 03272 000416         JMP RE,21       ;2ND FAILURE IN A ROW
12 03273 011126         ISZ ADDER,2     ;FIRST FAILURE
13 03274 020267 RE,20:  LDA 0,EVODN     ;IF TWO COMPUTERS
14 03275 101004         MOV 0,0,SZR     ;DON'T TRY AGAIN
15 03276 002273         JMP @RWRET      ;HEADS MAY HAVE MOVED
16 03277 006322         PCRLF
17 03300 006321         MESSAGE
18 03301 006231         MSG63           ;"TRY AGAIN"
19 03302 020245         LDA 0,RELRAN    ;RESET RAN # GEN FOR
20 03303 040244         STA 0,RANDOM    ;THE RETRY
21 03304 102400         SUB 0,0
22 03305 040202         STA 0,RETRY     ;CLEAR RETRY SWITCH
23 03306 062677         IORST
24 03307 002306         JMP @RLUP       ;DO IT AGAIN
25
26 03310 011132 RE,21:  ISZ PADER,2     ;COUNT PERM ADDR ERRS
27 03311 002273         JMP @RWRET      ;ERROR RETURN
28
29 03312 060477 RE,3:   READS 0         ;CHECK WORD ERROR
30 03313 101300         MOVS 0,0
31 03314 101202         MOVR 0,0,SZC    ;IF SW7=1 DON'T PRINT
32 03315 000405         JMP .+5
33 03316 006351         HEADER
34 03317 006322         PCRLF
35 03320 006321         MESSAGE
36 03321 006236         MSG64           ;"CHECK WORD ERROR"
37 03322 030174         LDA 2,UNIT
38 03323 010271         ISZ CWCNT       ;CHECK WORD ERR COUNTER
39 03324 000404         JMP RE,31       ;2ND ERROR IN A ROW
40 03325 011116         ISZ CWER,2      ;FIRST ERROR
41 03326 010202         ISZ RETRY       ;SET RETRY FLAG
42 03327 000403         JMP RE,32
43 03330 011122 RE,31:  ISZ CWERP,2     ;COUNT PERM CHK WD ERRS
44 03331 010203         ISZ FATAL       ;SET FATAL FLAG
45 03332 020254 RE,32:  LDA 0,CA        ;THE CHECK WORD ERROR MA
46 03333 065433         DIB 1,,DSKP     ;HAVE STOPPED THE DATA
47 03334 105400         SUB 0,1         ;TRANSFER PREMATURELY,
48 03335 020425         LDA 0,C174M     ;FIND THE # OF SECTORS
49 03336 123705         ANDS 1,0,SNR    ;TRANSFERRED AND SAVE IT
50 03337 102520         SUBZL 0,0       ;FOR THE DATA COMPARE RO TINE,
51 03340 040305         STA 0,,SC       ;IT MUST BE AT LEAST ONE
52 03341 000563         JMP .R1
53 03342 017400 C174M:  17400
```

```
;0059 .MAIN
01
02 03343 030174 RE.4:   LDA 2,UNIT      ;DATA ERROR
03 03344 020271          LDA 0,CWCNT
04 03345 010272          ISZ DACNT       ;DATA ERROR COUNTER
05 03346 000422          JMP RE.44       ;2ND DATA ERROR
06 03347 010202          ISZ RETRY       ;SET THE RETRY FLAG
07 03350 101404          INC 0,0,SZR     ;FIRST DATA ERROR
08 03351 000403          JMP RE.41       ;DATA AND CHECK WORD ERR R
09 03352 011136          ISZ DATER,2     ;ND CHK WD ERR
10 03353 000554          JMP .R2         ;GO ON
11 03354 024203 RE.41:   LDA 1,FATAL
12 03355 125004          MOV 1,1,SZR
13 03356 000405          JMP RE.43       ;2 CHECK WORD ERRORS
14 03357 015116          DSZ CWER,2      ;ONLY 1 CHECK WORD ERROR
15 03360 000401          JMP .+1         ;SKIPS SOMETIMES
16
17 03361 011142 RE.42:   ISZ CWDE,2      ;COUNT CHK WD & DATA ERR
18 03362 000545          JMP .R2         ;GO ON
19 03363 015122 RE.43:   DSZ CWERP,2     ;DON'T COUNT FATAL CHK W  ERR
20 03364 000401          JMP .+1
21 03365 102400          SUB 0,0         ;RESET FATAL ERROR FOR N W
22 03366 040203          STA 0,FATAL
23 03367 000772          JMP RE.42       ;TRY IT A 3RD TIME
24 03370 024203 RE.44:   LDA 1,FATAL     ;FATAL=NON-ZERO IF 2 CHK WD ERR
25 03371 125004          MOV 1,1,SZR
26 03372 000403          JMP RE.45       ;DATA & CHK WD ERR
27 03373 011152          ISZ PDER,2      ;DATA ERR ONLY
28 03374 000404          JMP RE.45       ;2ND ONE IS FATAL
29 03375 015122 RE.45:   DSZ CWERP,2     ;DISCOUNT PERM CHK WD ER
30 03376 000401          JMP .+1
31 03377 011146          ISZ CWDEP,2     ;COUNT PERM COMBO ERR
32 03400 010203 RE.46:   ISZ FATAL       ;SET THE FATAL FLAG
33 03401 000526          JMP .R2         ;GO ON
```

```
;0060 .MAIN
01
02 03402 004414 RE.5:   JSR SWCE         ;ENDING MEM ADDR ERR
03 03403 006172          MSG57           ;IDENTIFIER
04 03404 030174          LDA 2,UNIT
05 03405 011156          ISZ MISC,2
06 03406 010203          ISZ FATAL       ;IT'S FATAL
07 03407 000527          JMP .R3
08
09 03410 004406 RE.61:   JSR SWCE         ;ENDING DISK ADDR ERROR
10 03411 006277          MSG67           ;IDENTIFIER
11 03412 030174          LDA 2,UNIT
12 03413 011156          ISZ MISC,2      ;COUNT AS MISC TYPE ERR
13 03414 010203          ISZ FATAL       ;IT'S FATAL
14 03415 000556          JMP .R4
15
16                       ;SA + WC ERROR
17                       ;AC0 = BAD, AC1 = GOOD
18
19 03416 054423 SWCE:    STA 3,SWCRET
20 03417 035400          LDA 3,0,3       ;GET ERROR IDENTIFIER
21 03420 054405          STA 3,SWCE1
22 03421 006314          SAVAC
23 03422 006351          HEADER
24 03423 006322          PCRLF
25 03424 006321          MESSAGE
26 03425 000000 SWCE1:   0
27 03426 006322          PCRLF
28 03427 006321          MESSAGE
29 03430 006144          MSG51           ;"GOOD"
30 03431 024241          LDA 1,SAV1
31 03432 006317          TYPAC1
32 03433 006321          MESSAGE
33 03434 006147          MSG52           ;"BAD"
34 03435 024240          LDA 1,SAV0
35 03436 006317          TYPAC1
36 03437 010402          ISZ SWCRET
37 03440 002401          JMP @SWCRET
38 03441 000000 SWCRET:  0
39
40 03442 030174 FLWRT:   LDA 2,UNIT       ;GET UNIT DATA &
41 03443 011156          ISZ MISC,2      ;COUNT ANY WRITE ERROR
42 03444 040412          STA 0,ERW1      ;SAVE ERROR STATUS
43 03445 006322          PCRLF
44 03446 006321          MESSAGE
45 03447 006421          MSG76           ;"ERROR STATUS AFTER WRI E = ..."
46 03450 024406          LDA 1,ERW1      ;PRINT OUT
47 03451 006317          TYPAC1          ;ERROR STATUS WORD
48 03452 014310          DSZ XXWRT       ;COUNT DOWN # RETRIES
49 03453 002402          JMP @.+2
50 03454 002350          PSTAT
51 03455 003274          RE.20
52
53 03456 000000 ERW1:    0
54 03457 003312 XRE.3:   RE.3
```

```
10061 .MAIN
01
02                      ;READ SUBROUTINE
03                      ;RETURN+3, NORMAL
04                      ;RETURN+1, FATAL ERROR
05
06
07 03460 003476         RSRD            ;RETRY ADDRESS
08 03461 054273 .READ:  STA 3,RWRET
09 03462 020776         LDA 0,.READ-1   ;SET THE RETRY ADDRESS
10 03463 040326         STA 2,RLUP
11 03464 020023         LDA 0,C2        ;SET PROGRAM MODE
12 03465 040204         STA 0,MODE
13 03466 102000         ADC 0,2
14 03467 040272         STA 0,AECNT     ;SET RETRY COUNTERS
15 03470 040271         STA 2,CWCNT
16 03471 040272         STA 2,DACNT
17 03472 102400         SUB 0,0
18 03473 040202         STA 0,RETRY     ;CLEAR RETRY AND
19 03474 040203         STA 0,FATAL     ;FATAL FLAGS,
20 03475 040363         STA 0,NO.BS     ;CLEAR BAD SECTOR COUNTE
21 03476 006332 RSRD:   CLRRB           ;CLEAR BUFFER
22 03477 006333         SETP            ;SET PARAMETERS
23 03500 032174         LDA 2,UNIT
24 03501 021056         LDA 0,CYLT,2
25 03502 061133         DOAS 0,.DSKP    ;READ!!
26 03503 006356         INTWT           ;WAIT FOR INTERRUPT
27 03504 101011         SKIP
28 03505 002404         JMP .+4
29 03506 030174         LDA 2,UNIT      ;INCRMT THE READ/WRITE
30 03507 011156         ISZ RWTMO,2     ;TIMEOUTS COUNTER !!
31 03510 002273         JMP @RWRET      ;TIMEOUT
32
33 03511 101113         MOVL# 0,0,SNC   ;INTERRUPT RETURN
34 03512 002502         JMP @IRE1       ;NO R/W DONE FLAG
35 03513 024227         LDA 1,CSP       ;077662
36 03514 107404         AND 0,1,SZR
37 03515 002477         JMP @IRE1       ;FATAL STATUS
38 03516 024025         LDA 1,C10
39 03517 107404         AND 0,1,SZR
40 03520 002475         JMP @IRE2       ;ADDRESS ERROR
41 03521 024024         LDA 1,C4
42 03522 107404         AND 0,1,SZR
43 03523 002734         JMP @XRE,3      ;CHECK WORD ERROR
44
45 03524 006330 .R1:    CHECK           ;DATA COMPARE
46 03525 000000 RDATA:  0               ;ADDR OF DATA GEN ROUTIN
47 03526 002515         JMP RE,4        ;DATA ERROR
48
49 03527 061433 .R2:    DIB 0,.DSKP     ;READ ENDING ADDRESS
50 03530 024025         LDA 1,.SC       ;# SECTORS XFERRED
51 03531 125300         MOVS 1,1        ;# WORDS
52 03532 030254         LDA 2,CA        ;STARTING MEM ADDR
53 03533 147720         ADC 2,1         ;CORRECT ADDR
54 03534 106414         SUB# 0,1,SZR
55 03535 000645         JMP RE,5        ;ENDING MEM ADDR ERROR
```

```
10062 .MAIN
01
02 03536 020305 .R3:    LDA 0,.SC       ;FIND ENDING DISK ADDR
03 03537 040277         STA 0,CSC
04 03540 006353         JSR @ISCNT      ;AC0=SECT AC1=HEAD
05 03541 034305         LDA 3,.SC       ;IF .SC NOT = SC A CHK W  ERR
06 03542 030252         LDA 2,SC        ;TERMINATED THE DATA XFE
07 03543 172414         SUB# 3,2,SZR
08 03544 000426         JMP .+6         ;CHK WD ERR
09 03545 101004         MOV 0,0,SZR     ;IF SECT = 0 NO HEAD ICR
10 03546 002404         JMP .+4         ;OCCURRED AT THE END OF
11 03547 020211         LDA 0,LS        ;THE DATA TRANSFER
12 03550 124400         NEG 1,1         ;SECT = LAST SECTOR ON D SK
13 03551 124000         COM 1,1         ;HEAD = HEAD-1
14 03552 030174         LDA 2,UNIT
15 03553 035026         LDA 3,UNTINS,2
16 03554 125300         MOVS 1,1        ;FORM THE DISK ADDR
17 03555 167000         ADD 3,1         ;WORD AS READ VIA "DIC"
18 03556 103120         ADDZL 0,0
19 03557 103120         ADDZL 0,0       ; SECT*4
20 03560 107000         ADD 0,1
21 03561 020305         LDA 0,.SC       ;# SECT XFERRED
22 03562 030252         LDA 2,SC        ;# SECT SPECIFIED
23 03563 112400         SUB 0,2         ;AC2= # SECT NOT XFERRED
24 03564 020213         LDA 0,C17
25 03565 150400         NEG 2,2
26 03566 113400         AND 0,2         ;FINALLY AC0= EXPECTED
27 03567 147000         ADD 2,1         ;DISK ADDRESS
28 03570 062433         DIC 0,.DSKP     ;READ ACTUAL
29 03571 106414         SUB# 0,1,SZR
30 03572 000616         JMP RE,6        ;ENDING DISK ADDR ERROR
31
32 03573 030174 .R4:    LDA 2,UNIT      ;ADD #WORDS READ TO TOTAL
33 03574 020305         LDA 0,.SC
34 03575 101300         MOVS 0,0
35 03576 025112         LDA 1,#DSR+4,2  ;DOUBLE PRECISION
36 03577 107022         ADDZ 0,1,SZC
37 03600 011106         ISZ #DSR,2
38 03601 003401         NOP
39 03602 045112         STA 1,#DSR+4,2
40 03603 020223         LDA 0,FATAL     ;END OF TEST
41 03604 101004         MOV 0,0,SZR     ;WHAT TO DO NOW
42 03605 002273         JMP @RWRET      ;(FATAL) ERROR RETURN
43 03606 020202         LDA 0,RETRY
44 03607 101005         MOV 0,0,SNR
45 03610 002403         JMP @.+3        ;DO NORMAL RETURN
46 03611 002401         JMP @.+1        ;TRY AGAIN
47 03612 003274         RE,22
48 03613 003240         GENRET
49
50 03614 003262 IRE1:   RE,1
51 03615 003264 IRE2:   RE,2
```

```
;0063 .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 03616 054435 HSS:      STA 3,HSSRET
10 03617 006324           GETATM          ;GET "SAME", OR HEAD #
11 03620 034434           LDA 3,SAM       ;AC0=# AC1= NAME
12 03621 136415           SUB# 1,3,SNR
13 03622 002426           JMP HSS.1       ;"SAME"
14 03623 125003           MOV 1,1,SNC
15 03624 125004           MOV 1,1,SZR
16 03625 002343           JMP @IGCS       ;OTHER NAMES OR CR ILLEG L
17 03626 024210           LDA 1,LHD       ;CHECK HEAD LIMIT
18 03627 106437           SUBZ# 0,1,SBN
19 03630 002355           JMP @IGUST      ;LIMIT EXCEDED
20 03631 040253           STA 2,HD        ;HEAD #
21 03632 006324           GETATM          ;GET SECTOR #
22 03633 125003           MOV 1,1,SNC     ;AC0=# AC1=NAME
23 03634 125004           MOV 1,1,SZR
24 03635 002343           JMP @IGCS       ;NAME OR CR TERM ILLEGAL
25 03636 024211           LDA 1,LS        ;CHECK SECT # LIMIT
26 03637 106437           SUBZ# 0,1,SBN
27 03640 002355           JMP @IGUST      ;LIMIT EXCEDED
28 03641 040251           STA 2,SEC       ;SECTOR #
29 03642 006324           GETATM          ;GET # OF SECTORS
30 03643 101005           MOV 2,0,SNR     ;NO SECTOR COUNT,
31 03644 002355           JMP @IGUST
32 03645 125004           MOV 1,1,SZR
33 03646 002343           JMP @IGCS       ;NAME ILLEGAL
34 03647 040252           STA 2,SC        ;# OF SECTORS
35 03650 102550 HSS.1:    SUBCL 0,0
36 03651 040255           STA 2,TERM
37 03652 002421           JMP @HSSRET
38 03653 000000 HSSRET:   0
39 03654 046055 SAM:      45055           ;"SAME"
40 03655 000000 HIADD:    0               ;I/O ADDRESS MODIFICATIO , HI END
```

```
;0064 .MAIN
01
02
03                ;       CHECK SWITCH REGISTER ON ERROR,
04                ;       IF SW1 = 0  RETURN+1 (SCOPE LOOP)
05                ;       IF SW1 = 1  RETURN+2 (GO TO NEXT TEST)
06                ;       IF SW6 = 1  HALT AFTER ERROR
07
08 03656 060477 .CSW:     READS 0
09 03657 024047           LDA 1,C1000
10 03660 107404           AND 0,1 SZR     ;IS SW6 1?
11 03661 063077           HALT            ;YES,HALT
12 03662 062677           IORST           ;NO
13 03663 060477           READS 0         ;READ SWITCHES AGAIN IN  ASE
14 03664 103102           ADDL 0,0 SZC    ;THEY WERE CHANGED,
15 03665 175421           INC 3,3,SKP     ;NORMAL RETURN
16 03666 102521           SUBZL 0,0,SKP   ;SET LOOP SWITCH
17 03667 102400           SUB 0,0
18 03670 040205           STA 2,LUPSW
19 03671 001400           JMP 0,3
20
21
```

```
!0065 .MAIN
01                      /       THE DEVICE CODE CHANGE ROUTINE.......
02 03672 062677 DXCHG:  IORST           ;CLEAR THE WORLD...
03 03673 006322         PCRLF           /
04 03674 006321         MESSAGE
05 03675 006512         MSGD1
06 03676 024372         LDA 1,DEVICE    ;TYPES OUT THE PRESENT D VICE CODE,
07 03677 006320         TYPZ1           ;STORED IN LOCATION "DEV CE",
08 03700 006322 DXLP1:  PCRLF
09 03701 006321         MESSAGE         ;REQUESTS NEW DEVICE COD  OR CARRIAGE
10 03702 006527         MSGD2           ;RETURN IF PRESENT CODE  S OK,
11 03703 006434         JSR @DTIND      ;DTIND LOOKS FOR TTY INP T,
12 03704 000774         JMP DXLP1       ;BAD FINGER, TRY AGAIN,
13 03705 125035         MOVZ# 1,1,SNR   ;CKN FOR CR/W,O, CODE CH NGE,
14 03706 000415         JMP DXLP2       ;NO CHANGE IN DEV, CODE, ,
15 03707 020431         LDA 0,C76       ;CK IF CODE IS
16 03710 106453         SUBO# 0,1,SNC   ;GREATER THAN 76 OCTAL,
17 03711 000767         JMP DXLP1       ;TOO BIG, TRY AGAIN,
18 03712 030372         LDA 2,DEVICE
19 03713 132435         SUBZ# 1,2,SNR   ;CKN IF OLD CODE WAS TYP D IN,
20 03714 000407         JMP DXLP2       ;IF SO NO NEED TO CHANGE
21 03715 044421         STA 1,DVTMP     ;SAVE IN DEVICE TEMP,
22 03716 121000         MOV 1,0         ;SET AC0 WITH NEW CODE,
23 03717 004422         JSR DCHNG       ;GO CHANGE CODE,
24 03720 000372         DEVICE          ;CALL+1 ADDRS OF OLD COD ,
25 03721 000515         STRT            ;CALL+2 IS BEG OF PROGRA ,
26 03722 003655         HIADD           ;CALL+3 IS END OF PROGRA ,
27 03723 006322 DXLP2:  PCRLF
28 03724 006321         MESSAGE         ;PRINTS MESSAGE THAT NEW
29 03725 006564         MSGD3           ;DEV, CODE IS SET INTO T E
30 03726 024372         LDA 1,DEVICE    ;THE PROGRAM, ALSO PRINT  OUT
31 03727 006320         TYPZ1           ;THE OCTAL VALUE OF THE
32 03730 006322         PCRLF           ;NEW CODE....
33 03731 006321         MESSAGE         ;MESSAGE 4 IS A REMINDER TO
34 03732 006576         MSGD4           ;OPERATOR THAT THE PROGR M
35 03733 006322         PCRLF           ;MUST BE RESTARTED,
36 03734 063077         HALT
37 03735 000777         JMP .-1
38
39 03736 000000 DVTMP:  0
40 03737 004272 DTIND:  TIND
41 03740 000076 C76:    76
42
```

```
!0066 .MAIN
01
02                      ;CHANGE ALL DEVICE CODES FROM THE LOCATION IN CA L+2
03                      ;UP TO BUT NOT THE LOCATION IN CALL+3, FROM THE  LD CODE
04                      ;WHOSE ADDRESS IS IN CALL+1 TO WHAT IS IN AC2,
05                      ;IGNORES CODE 77, EXIT TO CALL+4,
06
07 03741 171400 DCHNG:  INC 3,2
08 03742 151400         INC 2,2
09 03743 050441         STA 2,DCH,5
10 03744 024441         LDA 1,DCH,1
11 03745 037376         LDA 3,@-2,2
12 03746 137400         AND 1,3
13 03747 057376         STA 3,@-2,2
14 03750 123400         AND 1,0
15 03751 040435         STA 0,DCH,6
16 03752 031377         LDA 2,-1,2
17 03753 136414         SUB# 1,3,SZR
18 03754 122415         SUB# 1,0,SNR
19 03755 000424         JMP DCH,2
20 03756 021000 DCH,4:  LDA 0,0,2
21 03757 103112         ADDL# 0,0,SZC   ;IS IT AN I/O INSTRUCTIO  ?
22 03760 101103         MOVL 0,0,SNC
23 03761 103113         ADDL# 0,0,SNC
24 03762 000412         JMP DCH,3       ;NO !!
25 03763 101200         MOVR 0,0
26 03764 102400         SUB 3,0
27 03765 123414         AND# 1,0,SZR
28 03766 000405         JMP DCH,3       ;NOY OLD DEVICE CODE,
29 03767 034417         LDA 3,DCH,6
30 03770 163000         ADD 3,0
31 03771 041000         STA 0,0,2
32 03772 034412         LDA 3,DCH,5
33 03773 037775         LDA 3,@-2,3
34 03774 151400 DCH,3:  INC 2,2
35 03775 022407         LDA 0,@DCH,5
36 03776 142414         SUB# 2,0,SZR
37 03777 000757         JMP DCH,4
38 04000 034406         LDA 3,DCH,6
39 04001 030403 DCH,2:  LDA 2,DCH,5
40 04002 057376         STA 3,@-2,2
41 04003 001001         JMP 1,2
42
43 04004 000000 DCH,5:  0
44 04005 000077 DCH,1:  77
45 04006 000000 DCH,6:  0
46
```

```
10067 .MAIN
01
02                ;AC1 REM AC0=(AC0,AC1)/AC2
03 04007 102400 DIVIO:  SUB     0,0
04 04010 054431 DIVDO:  STA     3,MSAV
05 04011 142432         SUBZ#   2,0,SZC
06 04012 002413         JMP     DEXT
07 04013 054426 DVYDO:  STA 3,MSAV       ;DIVIDE
08 04014 034425         LDA 3,M20
09 04015 125120         MOVZL 1,1
10 04016 101100 DLOOP:  MOVL 0,0
11 04017 142412         SUB# 2,0,SZC
12 04020 142400         SUB 2,0
13 04021 125100         MOVL 1,1
14 04022 175404         INC 3,3,SZR
15 04023 000773         JMP DLOOP
16 04024 176441         SUB0    3,3,SKP
17 04025 176420 DEXT:   SUBZ    3,3
18 04026 002413         JMP @MSAV
19
20                ;(AC0,AC1)=AC1*AC2+AC0
21 04027 102460 MULT:   SUBC 0,0          ;MULTIPLY
22 04030 054411 MULTA:  STA 3,MSAV
23 04031 034411         LDA 3,M20
24 04032 125203 MLOOP:  MOVR 1,1,SNC
25 04033 101201         MOVR 0,0,SKP
26 04034 143220         ADDZR 2,0
27 04035 175404         INC 3,3,SZR
28 04036 000774         JMP MLOOP
29 04037 125260         MOVCR 1,1
30 04040 002401         JMP @MSAV
31 04041 000000 MSAV:   0
32
33 04042 177760 M20:    -20
34
```

```
10068 .MAIN
01                ;TELETYPE NON INTERRUPT PACKAGE
02                ;AC1,AC2 SAVED
03                ;"MESS" PRINTS ASCII MESSAGES AS SPECIFIED BY AS EMBLER
04                ;"CRLF" PRINTS A CARRIAGE RETURN
05                ;"POCT" PRINTS C(1) IN OCTAL
06                ;"ZOCT" PRINTS C(1) IN OCTAL, LEADING ZEROS SUPP ESSED
07                ;"PDEC" PRINTS C(1) IN DECIMAL, LEADING ZEROS SU PRESSED,
08                ;THE ABOVE THREE ARE FOLLOWED BY THE TAB IN P,TA
09                ;"TIND" ACCEPTS OCTAL, AND
10                ;"TINO" ACCEPTS DECIMAL SINGLE PRECISION SIGNED  NTEGERS
11                ;INTO AC1 FROM THE TTI.  LEADING NULLS, TABS,
12                ;AND SPACES ARE IGNORED,  A 16 BIT UNSIGNED INTE ER IS
13                ;FORMED, THEN NEGATED IF A MINUS SIGN IS TYPED,
14                ;EXIT AT CALL+1 IF INPUT ERROR WITH AC0=BAD CHAR CTER,
15                ; (NOT A LEGAL DIGIT OR TERMINATING CHARACTER)
16                ;EXIT AT CALL+2 UPON TERMINATING CHARACTER
17                ; WITH AC0=0, 0, 40, 12, 55
18                ; FOR NULL, TAB, SPACE, CARRIAGE RETURN, COMMA
19                ;THE ABOVE WAIT FOR TTO DONE, THEN CLEAR TTO.
20                ;"CHAR" PRINTS ASCII CHARACTOR IN C(0)R; C(0)L M ST BE 0,
21                ;EXITS CALL +2 IF C(0)R=0, CORRECTS THE PARITY,
22                ;SIMULATES TAB ON ASR33.
23                ;"TYPE" PRINTS C(0)R,  MUST HAVE PROPER PARITY,  EXITS AT
24                ;CALL+1.  REPLACE "TYPE" WITH INTERRUPT TYPE IF  ESIRED.
25
26 04043 054554 MESS:   STA 3,MESSR      ;PRINT A TEXT MESSAGE
27 04044 044510         STA     1,P,AC1
28 04045 050510         STA     2,P,AC2
29 04046 010551         ISZ MESSR
30 04047 031400         LDA 2,0,3        ;C(2) POINTS TO MESSAGE
31 04050 024510         LDA 1,P,377      ;A 8 BIT MASK
32 04051 021000         LDA 0,0,2        ;C(2)=DATA WORD
33 04052 125112         MOVL# 1,1 SZC
34 04053 123701         ANDS 1,0 SKP
35 04054 123401         AND 1,0 SKP      ;C(0)=DATA CHARACTER RIG T
36 04055 151400         INC 2,2          ;INC TO NEXT WORD
37 04056 124000         COM 1,1          ;FLIP MASK
38 04057 004522         JSR CHAR         ;PRINT
39 04060 000771         JMP MESS+6       ;ANOTHER
40 04061 000402         JMP .+2
41 04062 004477 P,LST:  JSR CHAR
42 04063 024471 PEXIT:  LDA 1,P,AC1
43 04064 030471         LDA 2,P,AC2
44 04065 063511         SKPBZ TTO        ;NO,CLEAR TTO
45 04066 000777         JMP .-1
46 04067 063211         NIOC TTO
47 04070 063517         SKPBZ LPT
48 04071 000777         JMP .-1
49 04072 063217         NIOC LPT
50 04073 002524         JMP @MESSR       ;LAST
```

```
;0059 .MAIN
01 04074 102401 ZOCT:   SUB 0,0,SKP
02 04075 020462 POCT:   LDA     0,P,C60
03 04076 050457         STA     2,P,AC2
04 04077 030435         LDA 2,OCTAB     ;PRINT C(1) IN OCTAL
05 04100 000424         JMP .+4
06 04101 050454 PDEC:   STA     2,P,AC2
07 04102 030442         LDA 2,DECTB     ;PRINT C(1) IN DECIMAL
08 04103 102400         SUB 0,0
09 04104 054513         STA 3,MESSR     ;BOTH ENTRYS PRINT NUMBE
10 04105 044447         STA     1,P,AC1
11 04106 040445         STA 0,ZSUPP     ;THEN TAB TO NEXT POSITI N
12 04107 050401         STA 2,.+1
13 04110 000000 DECOCT: 0               ;A"LDA 2,TABLE" INSTRUCT ON
14 04111 010777         ISZ .-1
15 04112 020444         LDA 0,P,TAB
16 04113 151005         MOV 2,2,SNR     ;IF TABLE ENTRY=0
17 04114 000746         JMP P,LST       ;EXIT WITH  TAB
18 04115 034436         LDA 3,ZSUPP     ;ZEROS SUPPRESS STUF
19 04116 102400         SUB 0,0
20 04117 146452 DECOT:  SUBO# 2,1,SZC
21 04120 000425         JMP DECP
22 04121 146400         SUB 2,1         ;FORM THE DIGIT
23 04122 034435         LDA 3,P,C60
24 04123 101400         INC 0,0
25 04124 000773         JMP DECOT
26 04125 151235 DECP:   MOVZR# 2,2,SNR
27 04126 034431         LDA 3,P,C60
28 04127 054424         STA 3,ZSUPP     ;C(0)=DIGIT
29 04130 163000         ADD 3,0         ;MAKE ASCII
30 04131 175004         MOV 3,3,SZR
31 04132 004427         JSR CHAR        ;PRINT
32 04133 000755         JMP DECOCT      ;GET NEXT DIGIT
33
34 04134 030425 OCTAB:  LDA 2,.+1+.-DECOCT
35 04135 100000         100000
36 04136 010000         10000
37 04137 001000         1000
38 04140 000100         100
39 04141 000010         10
40 04142 000001         1
41 04143 000000         0
42 04144 030435 DECTB:  LDA 2,.+1+.-DECOCT
43       000012 .RDX 10
44 04145 023420         10000
45 04146 001750         1000
46 04147 000144         100
47 04150 000012         10
48 04151 000001         1
49 04152 000000         0
50       000010 .RDX 8
51 04153 000000 ZSUPP:  0
52 04154 000000 P,AC1:  0
53 04155 000000 P,AC2:  0
54 04156 000011 P,TAB:  11      ;CHARACTER PRINTED AFTER NUMBERS
55 04157 000060 P,C60:  60
56 04160 000377 P,377:  377
```

```
;0070 .MAIN
01 04161 054434 CHAR:   STA 3,CHRET     ;PRINT C(0) RIGHT
02 04162 101305         MOVS 0,0,SNR    ;RETURN +2 IF NULL
03 04163 001401         JMP 1,3
04 04164 115120         MOVZL   0,3     ;COMPUTE EVEN PARITY
05 04165 177004         ADD     3,3,SZR
06 04166 000777         JMP     .-1
07 04167 103200         ADDR    0,0
08 04170 101300         MOVS    0,0
09 04171 034463 CHAR1:  LDA     3,P,C11 ;IS THIS A TAB
10 04172 115415         SUB# 0,3,SNR
11 04173 000403         JMP CHA,3               ;YES
12 04174 004424         JSR TYPE        ;NO PRINT IT
13 04175 002420         JMP @CHRET      ;EXIT
14 04176 020461 CHA,3:  LDA     0,P,240 ;SIMULATE A TAB
15 04177 004421         JSR     TYPE    ;WITH 1 TO 14 SPACES
16 04200 020416         LDA     0,CHORZ
17 04201 034452         LDA     3,P,C7
18 04202 163404         AND     3,0,SZR
19 04203 000773         JMP     CHA,3
20 04204 040412         STA 0,CHORZ
21 04205 002410         JMP @CHRET
22
23 04206 054411 CRLF:   STA 3,MESSR     ;SAVE RETURN
24 04207 044745         STA     1,P,AC1
25 04210 050745         STA     2,P,AC2
26 04211 020445         LDA 0,P,C15
27 04212 004747         JSR CHAR        ;PRINT CARRIAGE AND LF
28 04213 020442         LDA     0,P,C12
29 04214 000646         JMP     P,LST
30
31 04215 000000 CHRET:  0
32 04216 000000 CHORZ:  0
33 04217 000000 MESSR:  0
```

```
10071 .MAIN
01 04220 054441 TYPE:   STA 3,TYPRET     ;TYPE THE C(0)R IF
02 04221 074477         READS 3          ;SWITCH 2(0).
03 04222 175100         MOVL 3,3
04 04223 177122         ADDZL 3,3,SZC
05 04224 000420         JMP TYPE2
06 04225 063511         SKPBZ TTO
07 04226 000777         JMP .-1
08 04227 061111         DOAS 0,TTO
09 04230 000414         JMP TYPE2
10 04231 034727 TYPE1:  LDA     3,P,377
11 04232 175220         MOVZR   3,3
12 04233 163400         AND     3,0
13 04234 116043         ADCO    0,3,SNC
14 04235 034423         LDA     3,P,C40
15 04236 162432         SUBZ#   3,0,SZC ;SKIP NON-PRINTING CHAR
16 04237 010757         ISZ     CHORZ
17 04240 034416         LDA     3,P,C15
18 04241 116445         SUBO    0,3,SNR
19 04242 054754         STA     3,CHORZ ;CLR HORZ POS
20 04243 002416         JMP @TYPRET
21 04244 175100 TYPE2:  MOVL 3,3
22 04245 177103         ADDL 3,3 SNC
23 04246 000763         JMP TYPE1
24 04247 063517         SKPBZ LPT
25 04250 000777         JMP .-1
26 04251 061117         DOAS 0,LPT
27 04252 000757         JMP TYPE1
28 04253 000007 P,C7:   7
29 04254 000011 P,C11:  11
30 04255 000012 P,C12:  12
31 04256 000015 P,C15:  15
32 04257 000240 P,240:  240
33 04260 000040 P,C40:  40
34 04261 000000 TYPRET: 0
```

```
10072 .MAIN
01 04262 020773 TINC,:  LDA     0,P,C12
02 04263 004735         JSR     TYPE
03 04264 010733 TINX,:  ISZ     MESSR
04 04265 024667 TINR,:  LDA     1,P,AC1
05 04266 034665         LDA     3,ZSUPP
06 04267 175102         MOVL    3,3,SZC
07 04270 124400         NEG     1,1
08 04271 002460         JMP     @PEX
09
10 04272 102121 TINO:   ADCZL   0,0,SKP ;OCTAL ENTRY
11 04273 102440 TIND:   SUBO    0,0     ;DECIMAL ENTRY
12 04274 054723         STA     3,MESSR
13 04275 050660         STA     2,P,AC2 ;AC2 IS SAVED
14 04276 030757         LDA     2,P,C12
15 04277 113020         ADD     0,2
16 04300 102440         SUBO    0,0
17 04301 040652         STA     0,ZSUPP ;MINUS SIGN AND LEADING  PACES FLAG
18 04302 034651 TINS,:  LDA     3,ZSUPP
19 04303 175004         MOV     3,3,SZR
20 04304 000760         JMP     TINX,
21 04305 054647 TINW,:  STA     3,P,AC1
22 04306 063610         SKPDN   TTI
23 04307 000777         JMP     .-1
24 04310 060610         DIAC    0,TTI
25 04311 004650         JSR     CHAR
26 04312 034746         LDA     3,P,C40
27 04313 116414         SUB#    0,3,SZR
28 04314 101015         MOV#    0,0,SNR
29 04315 000765         JMP     TINS,   ;SPACE, TAB, OR NULL
30 04316 024432         LDA     1,TIN2,
31 04317 106015         ADC#    0,1,SNR ;COMMA
32 04320 000744         JMP     TINX,
33 04321 106424         SUBZ    0,1,SZR ;MINUS
34 04322 000405         JMP     TINM,   ;NO
35 04323 034630         LDA     3,ZSUPP
36 04324 177200         ADDR    3,3     ;COMPLEMENT SIGN
37 04325 054626         STA     3,ZSUPP
38 04326 000760         JMP     TINW,+1
39 04327 136415 TINM,:  SUB#    1,3,SNR ;IS IT A CARRIAGE RETURN
40 04330 000732         JMP     TINC,
41 04331 024416 TINN,:  LDA     1,TIN1,
42 04332 107022         ADDZ    0,1,SZC ;SKIP IF NOT A DIGIT
43 04333 146513         SUBL#   2,1,SNC ;SKIP IF DIGIT
44 04334 000731         JMP     TINR,
45 04335 010616         ISZ     ZSUPP   ;OUT OF LEADING SPACES
46 04336 020616         LDA     0,P,AC1
47 04337 101120         MOVZL   0,0
48 04340 115120         MOVZL   0,3
49 04341 175120         MOVZL   3,3
50 04342 137000         ADD     1,3     ;8 OLD P,AC1'S + NEW DIG T
51 04343 145220         MOVZR   2,1
52 04344 125232         MOVZR#  1,1,SZC ;SKIP IF OCTAL MODE
53 04345 117000         ADD     0,3     ;ADD 2 OLD P,AC1'S
54 04346 000737         JMP     TINW,
55 04347 177720 TIN1,:  -60
56 04350 000055 TIN2,:  55
57 04351 004064 PEX:    PEXIT+1
```

```
10073 .MAIN
01                ;PROCESSOR TIMER PACKAGE
02                ;THIS PACKAGE IS CALLED WHENEVER IT IS NECESSARY TO
03                ;IDENTIFY THE MEAN TIME BASE OF THE COMPUTER IN
04                ;WHICH THE PROGRAM RESIDES, THE MEAN TIME BASE M Y
05                ;THEN BE UTILIZED TO VERIFY OR CALCULATE THE REL TIONSHIPS
06                ;OF OTHER PERIPHERAL FUNCTIONS.
07                ;
08                ;THE PACKAGE RETURNS TO THE CALL INSTRUCTION WIT
09                ;THE CONTENTS OF AC1= TO A CALIBRATION COUNT
10                ;THAT MAY BE INCREMENTED TO OVERFLOW IN 100 MILL -
11                ;SECONDS BY THE FOLLOWING DELAY LOOP.
12                ;TYPE1: MOV 0,0
13                ;       INC 0,0,SZC     ;SKP=NOT OVERFLOW
14                ;       JMP 1,3         ;EXIT LOOP
15                ;       0               ;ANY FLAVOR IO SKP
16                ;       JMP TYPE1
17                ;
18                ;THE CONTENTS OF AC2 WILL CONTAIN A SIMILAR 100  S
19                ;ITERATION COUNT BUT FOR THE FOLLOWING LOOP:
20                ;TYPE2: NIO 0
21                ;       DIA 1,,DEV      ;GET DEVICE STATUS
22                ;       AND# 2,1,SZR    ;ANY STATUS COMPARE
23                ;       JMP .+4         ;EXPECTED STATUS EXIT
24                ;       INC 0,0,SZR     ;SKP OUT ON LOOP OFLOW
25                ;       JMP TYPE2
26                ;
27                ;THE VALUES RETURNED MAY BE ARITHMETICALLY
28                ;PROCESSED (MULTIPLIED/DIVIDED) FOR LONGER OR
29                ;SHORTER DELAYS AS LONG AS THE STANDARD LOOPS
30                ;LISTED ABOVE ARE UTILIZED.
31                ;
32                ;IT IS RECOMMENDED THAT ALL TIMING FUNCTIONS
33                ;BE PERFORMED WITHIN THE SAME GENERAL AREA OF
34                ;MEMORY AS THIS TIMING PACKAGE.
```

```
10074 .MAIN
01                ;
02                ;INITIALLY, THE TIMER PACKAGE ATTEMPTS TO
03                ;DETERMINE IF THE COMPUTER HAS A REAL TIME CLOCK
04                ;AVAILABLE, THE ITERATION COUNTS ARE DEVELOPED S MPLY
05                ;BY SYNCING WITH THE CLOCK AND COUNTING
06                ;THE NOF LOOP ITERATIONS AT 10 HZ.
07                ;
08                ;IF, HOWEVER THERE IS NO REAL TIME CLOCK THE MEA
09                ;TIME BASE OF THE LOOPS MUST BE CALCULATED, THIS
10                ;IS PERFORMED BY COUNTING THE NOF TIMES THE
11                ;STANDARD LOOPS ARE ITERATED FOR ONE OUTPUT
12                ;CHARACTER TO DEVICE "TTO" AND REQUESTING THE
13                ;BAUD RATE OF DEVICE TTO BE TYPED IN BY THE
14                ;TEST OPERATOR.
15                ;
16 04352 054557 PTIME:  STA 3,SVTIME
17 04353 063514         SKPBZ RTC       ;TEST FOR CAS/RTC
18 04354 000416         JMP SCORE       ;CAS RTC NONEXIS,
19 04355 060114         NIOS RTC        ;TURN CLOCK ON
20 04356 063514         SKPBZ RTC       ;BUSY =1 IS RTC
21 04357 000423         JMP .+3
22 04360 063614         SKPDN RTC       ;DONE =0 NO RTC
23 04361 000411         JMP SCORE       ;AND DEV TTO IS USED
24 04362 062677         IORST
25 04363 102520         SUBZL 0,0       ;=10 HZ FOR RTC
26 04364 004473         JSR TYME
27 04365 061114         DOAS 0,RTC      ;PASSED TO "TYME"
28 04366 063514         SKPBZ RTC       ;FOR EXECUTION
29 04367 002542         JMP @SVTIME     ;AC1 AND AC2=LOOP COUNTS
30                ;
31                ;THE FOLLOWING SUBROUTINE IS TO RETAIN COMPATABI ITY
32                ;WITH THE OLD PROCESSOR IDENTIFICATION AND TIMIN
33                ;PACKAGE TO RETRIEVE THE LOOP COUNT FOR
34                ;THE DIA,B OR C LOOP TYPE 2
35 04370 024540 .3DI:   LDA 1,NUCAL
36 04371 001400         JMP 0,3
```

```
10075 .MAIN
01                      ;
02                      ;THERE IS NO RTC-UTILIZE DEVICE TTO AND ASK FOR
03                      ;BAUD RATE INPUT FROM OPERATOR
04 04372 062677 SCORE:  IORST
05 04373 102400         SUB 0,0         ;AC0=NULL CHARACTER
06 04374 004463         JSR TYME
07 04375 061111         DOAS 0,TTO      ;PASSED TO TYME
08 04376 063511         SKPBZ TTO       ;FOR EXECUTION
09 04377 006533 SCORA1: JSR @TUMBLER    ;OUT TEXT
10 04400 004542         SESOUT          ;ASKING FOR BAUD RATE
11                      ;THE FOLLOWING SERIES OF INSTRUCTIONS WILL
12                      ;CALCULATE THE ITERATION COUNT FOR
13                      ;1 BIT OF TTO OUTPUT AFTER RETRIEVING
14                      ;THE CONSOLE BAUD RATE FROM THE
15                      ;TEST OPERATOR---REQUIRES SUBROUTINE TIND
16 04401 006533         JSR @KEYS
17 04402 002775         JMP SCORA       ;INPUT ERROR
18 04403 044530         STA 1,LOCK      ;SAVE BAUD RATE
19 04404 030535         LDA 2,S,3D1     ;10
20                      ;ROUTINE ASSUMES AN 11 BIT CHARACTER
21 04405 151400         INC 2,2         ;ASSUME 11 BITS
22 04406 024521         LDA 1,ORDINAL   ;COUNT FOR FULL CHAR
23 04407 102400         SUB 0,0
24 04410 006526         JSR @KEYS+2     ;CHAR TIME/#BITS
25 04411 101004         MOV 0,0,SZR     ;IF ANY REM,
26 04412 125400         INC 1,1         ;FUDGE BIT COUNT
27 04413 020520         LDA 0,LOCK
28 04414 044517         STA 1,LOCK      ;SAVE ITR COUNT 1 BIT
29 04415 131000         MOV 1,2
30 04416 105000         MOV 0,1         ;AC1 = BAUD RATE
31                      ;BAUD RATE TIMES COUNT FOR 1 BIT
32                      ;WILL EQUAL ITERATION COUNT FOR 1 SECOND
33 04417 102400         SUB 0,0
34 04420 006515         JSR @KEYS+1     ;MUL AC1*AC2
35 04421 040517         STA 0,KN        ;SAVE DOUBLE LENGTH
36 04422 044515         STA 1,KS        ;RESULT
37                      ;1 SECOND DIVIDED BY 10 = 100 MILLISECONDS
38 04423 030516         LDA 2,S,3D1     ;10
39 04424 006512         JSR @KEYS+2
40 04425 030502         LDA 2,ORDINAL   ;COUNT FOR 1 CHAR
41 04426 044501         STA 1,ORDINAL   ;ORDINAL=100 MS TYPE 1
42                      ;CONTINUE CALCULATIONS NEXT PAGE
```

```
10076 .MAIN
01                      ;
02                      ;1 SECOND COUNT/1 CHAR COUNT = # CHAR'S PER SEC
03                      ;THIS CALCULATION IS USED TO EXPAND THE TYPE 2 C UNT
04 04427 020511         LDA 0,KN
05 04430 024507         LDA 1,KS        ;1 SEC, RESTORED
06 04431 006505         JSR @KEYS+2     ;DIVIDE BY CHAR,
07 04432 044505         STA 1,KS        ;# CHAR, 1 SEC
08                      ;CALC RELATIONSHIP OF REM, TO 1 CHAR TO FILL SEC ND
09 04433 145120         MOVZL 2,1
10 04434 111005         MOV 0,2,SNR
11 04435 151400         INC 2,2
12 04436 102400         SUB 0,0
13 04437 006477         JSR @KEYS+2     ;DIVIDE REM INTO CHAR
14                      ;AC1=FUDGE FACTOR 1 RELEATIONSHIP OF CHAR TO TOT L 1 SEC
15                      ;FINISH CALCULATIONS ON LOOP TYPE 1 TO= 1SECOND
16 04440 131000         MOV 1,2         ;FUDGE FACTOR
17 04441 024467         LDA 1,NUCAL     ;INTO CHARACTER TIME
18 04442 125120         MOVZL 1,1
19 04443 102400         SUB 0,0         ;WILL =
20 04444 006472         JSR @KEYS+2     ;PORTION OF CHAR
21 04445 121000         MOV 1,0         ;TO COMPLETE 1 SECOND
22 04446 024462         LDA 1,NUCAL     ;1 CHAR, TYPE 2 LOOP
23 04447 030470         LDA 2,KS        ;# CHARS IN 1 SEC
24 04450 006465         JSR @KEYS+1     ;+ PORTION OF CHAR
25                      ;DOUBLE LENGTH AC0,AC1=1 SECOND FOR TYPE 2 LOOP
26 04451 030470         LDA 2,S,3D1     ;DIVIDE BY 10 FOR 100 MS
27 04452 006464         JSR @KEYS+2
28 04453 044455         STA 1,NUCAL
29 04454 131000         MOV 1,2         ;AC2=100MS LOOP2
30 04455 024452         LDA 1,ORDINAL   ;AC1 =100MS LOOP 1
31 04456 002453         JMP @SVTIME
```

```
10077 .MAIN
01                      ;
02                      ;THE FOLLOWING SUBROUTINE DETERMINES THE ITERATI N
03                      ;COUNTS FOR THE DEVICE SPECIFIED BY THE INSTRUCT ONS
04                      ;FOLLOWING THE JSR CALL TO TYME
05                      ;DOAS 0,RTC     OR      DOAS 0,TTO
06                      ;SKPBZ RTC      OR      SKPBZ TTO
07                      ;
08 04457 024001 TYME:   LDA 1,1         ;SAVE INTR, LINK
09 04460 044446         STA 1,RVTMP
10 04461 024437         LDA 1,ENTYM
11 04462 044001         STA 1,1         ;FOR LOOP 2 INTR,
12 04463 025400         LDA 1,0,3       ;GET DOAS
13 04464 044410         STA 1,TIMA
14 04465 044414         STA 1,TIMB
15 04466 044421         STA 1,TIMC      ;FOR EXECUTE
16 04467 025401         LDA 1,1,3       ;GET SKPBZ
17 04470 044405         STA 1,TIMA+1
18 04471 044414         STA 1,TIMB+4    ;FOR EXECUTE
19 04472 152400         SUB 2,2
20 04473 126400         SUB 1,1         ;CLR CTRS
21 04474 061114 TIMA:   DOAS 0,RTC      ;OR TTO
22 04475 063514         SKPBZ RTC
23 04476 000777         JMP .-1         ;WAIT FOR DONE
24 04477 124004         COM 1,1,SZR     ;AND 2ND DONE
25 04500 000774         JMP TIMA        ;THEN START COUNTING
26 04501 061114 TIMB:   DOAS 0,RTC      ;THE THIRD DONE
27                      ;THE FOLLOWING COMPRISES LOOP TYPE 1
28 04502 101000         MOV 0,0
29 04503 125405         INC 1,1,SNR     ;WATCH FOR OFLOW
30 04504 000403         JMP .+3
31 04505 063514         SKPBZ RTC
32 04506 000774         JMP TIMB+1
33                      ;LOOP TYPE 2 IS COUNTED UNTIL PI FROM DEVICE
34 04507 061114 TIMC:   DOAS 0,RTC
35 04510 060177         INTEN
36                      ;THE FOLLOWING INSTR, COMPRISE THE LOOP TYPE 2
37 04511 050000         NIO 0   ;AND IT ITERATES UNTIL
38 04512 060400         DIA 0,0 ;INTERRUPTED BY PI
39 04513 102005         ADC 0,0,SNR
40 04514 004373         JSR HL          ;FILL INSTR,
41 04515 151404         INC 2,2,SZR     ;LOOP CTR
42 04516 000773         JMP TIMC+2
43 04517 004373         JSR HL  ;DEVICE OR PI FAILED
44 04520 004521 ENTYM:  .+1     ;TO HERE WHEN PI
45 04521 044406         STA 1,ORDINAL   ;SAVE LOOP 1
46 04522 050406         STA 2,NUCAL     ;AND LOOP 2
47 04523 020403         LDA 0,RVTMP
48 04524 040001         STA 0,1         ;RESTORE INTR, LINK
49 04525 001402         JMP 2,3 ;RETURN TO CALL
```

```
10078 .MAIN
01                      ;
02                      ;CONSTANTS SUBR, LINKS AND TEMP STORES
03 04526 000000 RVTMP:  0
04 04527 000000 ORDINAL:        0
05 04530 000000 NUCAL:  0
06 04531 000000 SVTIME: 0
07 04532 004043 TUMBLER:        MESS
08 04533 000000 LOCK:   0
09 04534 004273 KEYS:   TIND
10 04535 004030         MULTA
11 04536 004010         DIVDC
12 04537 000000 KS:     0
13 04540 000000 KN:     0
14 04541 000012 S.3D1:  10.
15 04542 005215 SESOUT: .TXTE |<15><12><12>
16 04543 152012 TTO BAUD RATE ?= |
17              147724
18              041240
19              052501
20              120104
21              040722
22              142724
23              037540
24              120275
25              000000
```

```
10079 .MAIN
01
02                      ;INITIALIZING SUBROUTINE FOR SUBROUTINE  LTYM,
03                      ;PREREQUISITES ARE:   THE PTIME PACKAGE  UST HAVE
04                      ;BEEN RUN SUCCESSFULLY AND THE CALIBRATE  COUNT
05                      ;RETURNED IN AC1 MUST BE STORED IN THE L CATION
06                      ;TAGGED "CALIB". LAYUP LOADS THE NEGATED COUNT FOR
07                      ;1 MILLISECOND INTO .WERK (OF THE DLTYM  UBR.) AND
08                      ;INTO AC1. ALL AC'S EXCEPT AC1 ARE SAVED  AC1 (C)
09                      ;RETURNED ZERO INDICATES "CALIB" NOT LCA ED.
10
11 04555 040423 LAYUP:  STA 0,DLY,1      ;SAVE OUR AC'S.
12 04556 050423          STA 2,DLY,2
13 04557 054423          STA 3,DLY,3
14 04560 102440          SUBO 0,0         ;CLEAR AC0 FOR DIVIDE.
15 04561 024415          LDA 1,CALIB      ;CALIB LOADED PREVIOUSLY BY MAIN.
16 04562 125015          MOVM 1,1,SNR     ;CKN IF (C) CALIB = 0.
17 04563 000407          JMP .+7
18 04564 030417          LDA 2,CXL,V
19 04565 006412          JSR @DLY,0       ;DIVIDE COUNT BY 100.
20 04566 124400          NEG 1,1
21 04567 046402          STA 1,@.+2
22 04570 000402          JMP .+2
23 04571 004634          .WERK
24 04572 020405          LDA 0,DLY,1      ;RESTORE AC'S AND RETURN WITH
25 04573 030405          LDA 2,DLY,2      ;1 MS COUNT IN AC1. (C)  C1 WILL
26 04574 034405          LDA 3,DLY,3      ;BE RETURNED 0 IF CALIB  S
27 04575 001400          JMP 0,3          ;NOT LOADED.
28
29 04576 000000 CALIB:   0
30 04577 004635 DLY,0:   DIVD
31 04600 000000 DLY,1:   0
32 04601 000000 DLY,2:   0
33 04602 000000 DLY,3:   0
34 04603 000144 CXL,V:   144
35
```

```
10080 .MAIN
01                      ;MAIN PROGRAM SHOULD BE CODED SUCH THAT  HIS SUBR.
02                      ;WILL BE CALLED BY "DELAY" FOR STANDARD  ORMAT.
03                      ;BOTH THE PTIME (PROCESSOR TIMER) PACKAG  AND
04                      ;SUBROUTINE LAYUP MUST HAVE BEEN CALLED  UCCESSFULLY
05                      ;PRIOR TO CALLING THIS SUBROUTINE, THE A GUMENT FOLLOW-
06                      ;ING THE CALL IS THE ADDRESS OF THE DELA  CONSTANT.
07                      ;ALL AC'S ARE SAVED, RETURNS TO CALL+2 O  END OF
08                      ;DELAY, A SAMPLE INSTRUCTION SEQUENCE SH WING THE
09                      ;PROPER LINKING OF ALL THREE SUBROUTINES IS AS
10                      ;FOLLOWS:
11                              ;JSR @STIME      ;WHERE STIME:    TIME.
12                              ;MOVM 1,1,SNR    ;VERIFY COUNT IS LOADED.
13                              ;HALT            ;NO COUNT,HALT O  ERROR.
14                              ;STA 1,@ICALIB   ;WHERE ICALIB:   ALIB.
15                              ;JSR @IXLAY      ;WHERE IXLAY:    AYUP.
16                              ;MOVM 1,1,SNR    ;VERIFY .WERK IS LOADED.
17                              ;HALT            ;NO COUNT HALT O  ERROR.
18                      ;*** PROGRAM CONTINUES TO POINT WHERE DE AY IS REQ'D.**
19                              ;DELAY           ;WHERE DELAY=JSR IDLY,AND
20                                               ;IDLY:  DLTYM.
21                              ;.25MS           ;WHERE .25MS:    5, ON PAGE 0.
22                      ;*** CONTINUE PROGRAM ..... * *
23
24
25                      ;PROGRAMMING OVERHEAD SHOULD BE FACTORED INTO ALL
26                      ;PRECISE DELAY MEASUREMENTS AND ARE DEFI ED AS
27                      ;FOLLOWS:.......
28                      ;NOVA COMPUTER(FROM CALL TO RETURN)..... .
29                      ;       54 MICROS PLUS 16.4 MICROS PER M LLISEC.
30                      ;1200 COMPUTER(FROM CALL TO RETURN).....
31                      ;       24.4 MICROS PLUS 5.4 MICROS PER  ILLISEC.
32                      ;800 COMPUTER(FROM CALL TO RETURN)......
33                      ;       15.2 MICROS PLUS 3.2 MICROS PER  ILLISEC.
34                      ;SUPER NOVA COMPUTER(FROM CALL TO RETURN ......
35                      ;       15.8 MICROS PLUS 3.2 MICROS PER  ILLISEC.
36                      ;SUPER NOVA SC COMPUTER(FROM CALL TO RET RN)...
37                      ;       11.7 MICROS PLUS 1.8 MICROS PER  ILLISEC.
38                      ;PROGRAMMING OVERHEAD DATA MAY ALSO BE U ED TO
39                      ;CALCULATE DELAY ACCURACY.
40
```

```
10081 .MAIN
01 04524 040425 DLTYM:  STA 0,DLY,4      ;SAVE OUR AC'S.
02 04525 044425         STA 1,DLY,5
03 04526 050425         STA 2,DLY,6
04 04527 152440         SUBO 2,2
05 04530 050235         STA 2,CMEGA      ;COUNTS MILLISECS
06 04531 031400         LDA 2,3,3        ;GET # MILLISECS ARGUMEN ,
07 04532 150400         NEG 2,2          ;AND NEGATE.
08 04533 024421         LDA 1,.MERK      ;(C),MERK = NEG COUNT FO  1MS.
09 04534 121000 DLTY1:  MOV 1,0
10 04535 101000         MOV 0,0          ;** THESE
11 04536 101405         INC 0,0,SNR      ;       ** INSTRUCTIONS  ORM
12 04537 000403         JMP .+3
13 04540 063620         SKPDN 0          ;          ** THE DELAY
14 04541 000774         JMP .-4          ;             ** TIMING LOOP,
15 04542 010235         ISZ CMEGA
16 04543 151424         INC 2,2,SZR      ;INC DOWN MULTIPLY COUNT R
17 04544 000770         JMP DLTY1        ;LOOP FOR NX MILLISEC,
18 04545 020424         LDA 0,DLY,4      ;RESTORE AC'S
19 04546 024424         LDA 1,DLY,5
20 04547 030424         LDA 2,DLY,6
21 04550 001401         JMP 1,3          ;AND RETURN,
22
23 04551 000000 DLY,4:  0
24 04552 000000 DLY,5:  0
25 04553 000000 DLY,6:  0
26 04554 000000 .MERK:  0
27
28
29              ;       AC0=(AC0,AC1)/AC2 , AC1 = REM
30 04555 054414 DIVO:   STA 3,MDV,5      ;DIVIDE
31 04556 034414         LDA 3,MDV,1
32 04557 125120         MOVZL 1,1
33 04560 121100 MDV,2:  MOVL 0,0
34 04561 142412         SUB# 2,0,SZC
35 04562 142400         SUB 2,0
36 04563 125100         MOVL 1,1
37 04564 175404         INC 3,3,SZR
38 04565 000773         JMP MDV,2
39 04566 176441         SUBO 3,3,SKP
40 04567 176420 MDV,3:  SUBZ 3,3
41 04570 002401         JMP @MDV,5
42 04571 000020 MDV,5:  0
43 04572 177760 MDV,1:  -20
44
45
46
47
```

```
10082 .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 04573 031400 SRH:    LDA 2,0,3        ;AC2=TABLE ADDRESS
10 04574 021000         LDA 0,0,2
11 04575 101005         MOV 0,0,SNR
12 04576 001401         JMP 1,3          ;NOT FOUND
13 04577 122415         SUB# 1,0,SNR
14 04600 001402         JMP 2,3          ;FOUND
15 04601 151400         INC 2,2
16 04602 000772         JMP SRH+1
17 04603 170000 C170K:  170000
18
19
20                      ;SIZE MEMORY
21
22 04604 054441 SM:     STA 3,SM,5       ;SAVE RETURN
23 04605 032247         LDA 2,LAST       ;GET END PROGRAM ADDRESS
24 04606 020223         LDA 0,C212       ;LOADER BUFFER
25 04607 035000 SM,1:   LDA 3,0,2        ;SAVE (C) OF LOCATION
26 04610 051000         STA 2,0,2        ;WHILE IT IS VERIFIED
27 04611 025000         LDA 1,0,2        ;LOCATION EXISTS,
28 04612 055000         STA 3,0,2        ;RESTORE (C) OF LOCATION
29 04613 146414         SUB# 2,1,SZR     ;CHECKING IF LOCATION EX STS
30 04614 000404         JMP SM,2         ;NO, 1 STEP BEYOND B
31 04615 151400         INC 2,2          ;YES, INC TO NX,
32 04616 151133         MOVZL# 2,2,SNC   ;CHECK FOR < 32K
33 04617 000770         JMP SM,1         ;STILL UNDER 32K MAX,
34 04620 112400 SM,2:   SUB 0,2          ;DEDUCT BUFFER FOR BIN,  OR,
35 04621 050265         STA 2,CMEND      ;SAVE END OF MEMORY,
36 04622 020247         LDA 0,LAST
37 04623 112420         SUB 0,2
38 04624 020757         LDA 0,C170K      ;HOW MANY SECTORS WILL M M
39 04625 143414         AND# 2,0,SZR     ;ALLOW
40 04626 000413         JMP SM,4         ;MAXIMUM, 20 OCTAL
41 04627 020226         LDA 0,C7400
42 04630 113700         ANDS 0,2         ;ROOM FOR 17 OR LESS
43 04631 050266         STA 2,SMAX       ;MAX SECTOR COUNT
44 04632 126400         SUB 1,1
45 04633 125140         MOVOL 1,1        ;GENERATE # SECTORS
46 04634 151224         MOVZR 2,2,SZR    ; MASK
47 04635 000776         JMP .-2
48 04636 125300 SM,3:   MOVS 1,1
49 04637 044304         STA 1,LMSK
50 04640 002405         JMP @SM,5
51
52
53 04641 024213 SM,4:   LDA 1,C17
54 04642 121400         INC 1,0
55 04643 040266         STA 0,SMAX       ;MAX SECT CNT=20
56 04644 000772         JMP SM,3
57 04645 000000 SM,5:   0
58
```

```
!0083 .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 04726 020510 GA.0:   LDA 0,FIND
11 04727 101025         MOV 0,0,SNR
12 04730 000413         JMP GA.1        ;NO NAME OR # YET
13 04731 020504         LDA 0,LINCT
14 04732 040502         STA 0,SLNCT     ;REMEMBER LINCT
15 04733 040502         STA 0,SRCH      ;SET SEARCH FLAG
16 04734 000407         JMP GA.1
17
18 04735 054474 GATM:   STA 3,GARET
19 04736 102400         SUB 0,0
20 04737 040473         STA 0,ANAM
21 04740 040473         STA 0,ANUM
22 04741 040474         STA 0,SRCH
23 04742 040474         STA 0,FIND
24 04743 030250 GA.1:   LDA 2,LINCT
25 04744 010250         ISZ LINCT
26 04745 151220         MOVZR 2,2
27 04746 021000         LDA 0,0,2
28 04747 024225         LDA 1,C377
29 04750 101002         MOV 0,0,SZC
30 04751 101300         MOVS 0,0
31 04752 123400         AND 1,0         ;(AC0)R = BYTE
32 04753 024216         LDA 1,C40
33 04754 122415         SUB# 1,0,SNR
34 04755 000751         JMP GA.0        ;SPACE DELIMITER
35 04756 024462         LDA 1,C54
36 04757 122415         SUB# 1,0,SNR
37 04760 000746         JMP GA.0        ;COMMA DELIMITER
38 04761 024224         LDA 1,C215
39 04762 122415         SUB# 1,0,SNR
40 04763 000442         JMP EXIT        ;CR DELIMITER
41 04764 024451         LDA 1,SRCH
42 04765 125004         MOV 1,1,SZR
43 04766 000434         JMP EX.1        ;SEARCH FLAG ON
44 04767 030217         LDA 2,C60
45 04770 034220         LDA 3,C67
46 04771 162033         ADCZ# 3,0,SNC
47 04772 112032         ADCZ# 0,2,SZC
48 04773 000412         JMP ASSN        ;NOT #
49 04774 024445         LDA 1,C7
50 04775 123400         AND 1,0
51 04776 024435         LDA 1,ANUM
52 04777 127120         ADDZL 1,1       ;ASSEMBLE OCTAL #
53 05000 125120         MOVZL 1,1
54 05001 123000         ADD 1,0
55 05002 040431         STA 0,ANUM
56 05003 010433         ISZ FIND
57 05004 000737         JMP GA.1        ;GET MORE
```

```
!0084 .MAIN
01
02 05005 024215 ASSN:   LDA 1,C37       ;ASSEMBLE NAME
03 05006 123400         AND 1,0
04 05007 024423         LDA 1,ANAM
05 05010 030427         LDA 2,C176K     ;3 LETTERS YET ?
06 05011 133404         AND 1,2,SZR
07 05012 000731         JMP GA.1        ;YES, IGNORE THE REST
08 05013 127120         ADDZL 1,1
09 05014 127120         ADDZL 1,1       ;5 LEFT
10 05015 125120         MOVZL 1,1
11 05016 123000         ADD 1,0
12 05017 040413         STA 0,ANAM
13 05020 010416         ISZ FIND
14 05021 000722         JMP GA.1        ;GET MORE
15
16 05022 020412 EX.1:   LDA 0,SLNCT
17 05023 040250         STA 0,LINCT
18 05024 101021         MOVZ 0,0,SKP    ;CLEAR CARRY
19 05025 101040 EXIT:   MOVO 0,0        ;SET CARRY
20 05026 020405         LDA 0,ANUM
21 05027 024403         LDA 1,ANAM
22 05030 002401         JMP @GARET
23
24 05031 000000 GARET:  0
25 05032 000000 ANAM:   0
26 05033 000000 ANUM:   0
27 05034 000000 SLNCT:  0
28 05035 000000 SRCH:   0
29 05036 000000 FIND:   0
30 05037 176000 C176K:  176000
31 05040 000054 C54:    54
32 05041 000007 C7:     7
33
```

```
!0085 .MAIN
01
02                      ;PRINT HEADER
03
04 05042 010172 HED:    ISZ HSW          ;PRINT HEADER ONLY ONCE
05 05043 021400          JMP 0,3          ;FOR EACH COMPLETE TEST
06 05044 054442          STA 3,HRET
07 05045 020442          LDA 2,HRET+1     ;THE MODE DETERMINES
08 05046 032204          LDA 2,MODE       ;THE TYPE OF OPERATION
09 05047 113020          ADD 0,2          ;IN PROGRESS
10 05050 021000          LDA 0,0,2
11 05051 040424          STA 2,HED.1
12 05052 006322          PCRLF
13 05053 006322          PCRLF
14 05054 006321          MESSAGE
15 05055 000000 HED.1:   0                ;"RECAL OR SEEK, ETC.
16 05056 006321          MESSAGE
17 05057 005127          MSG44            ;CYL-
18 05060 032174          LDA 2,UNIT
19 05061 025056          LDA 1,CYLT,2
20 05062 125415          INC# 1,1,SNR
21 05063 125400          INC 1,1
22 05064 006320          TYPZ1
23 05065 006321          MESSAGE
24 05066 006132          MSG45            ;HEAD-
25 05067 024253          LDA 1,HD
26 05070 006320          TYPZ1
27 05071 006321          MESSAGE
28 05072 006135          MSG46            ;SECT-
29 05073 024251          LDA 1,SEC
30 05074 006320          TYPZ1
31 05075 006321          MESSAGE
32 05076 006140          MSG47            ;#SECT-
33 05077 024252          LDA 1,SC
34 05100 006320          TYPZ1
35 05101 006321          MESSAGE
36 05102 005721          MSG17            ;UNIT:
37 05103 024174          LDA 1,UNIT
38 05104 006320          TYPZ1
39 05105 002401          JMP #HRET
40 05106 000000 HRET:    0
41 05107 005110          .+1
42 05110 005644          MSG2             ;0  RECAL
43 05111 005650          MSG3             ;1  SEEK
44 05112 006073          MSG32            ;2  READ
45 05113 006076          MSG33            ;3  WRITE
```

```
!0086 .MAIN
01
02
03
04 05114 102541 C:       SUBCL 0,0,SKP    ;CHECK DATA...
05 05115 102520 G:       SUBZL 0,0        ;GENERATE DATA...
06 05116 040514          STA 0,FSTGC      ;C(CA)=START ADDRESS
07 05117 054511          STA 3,GRET       ;C(SC)  =NUMBER OF SECTO S
08 05120 021400          LDA 2,0,3        ;WORD FOLLOWING THE CALL
09 05121 040512          STA 0,PATT       ;DEFINES THE PATTERN.
10 05122 010506          ISZ GRET
11 05123 020260          LDA 0,VARST
12 05124 040263          STA 0,VARPT
13 05125 020254          LDA 0,CA
14 05126 040020          STA 0,IDX0
15 05127 014020          DSZ IDX0
16 05130 020305          LDA 0,.SC
17 05131 101300          MOVS 0,0
18 05132 100400          NEG 0,0
19 05133 040501          STA 0,GWC
20 05134 014476          DSZ FSTGC        ;GEN/CHECK FIRST SWITCH
21 05135 000446          JMP CC
22 05136 006475 GG:      JSR #PATT        ;GENERATE DATA
23 05137 042020          STA 0,#IDX0      ;STORE IN MEMORY
24 05140 010474          ISZ GWC
25 05141 000775          JMP GG
26 05142 002466          JMP #GRET
27
28
29 05143 070477 CKER:    READS 2          ;IF SW7=1 DON'T PRINT ER ORS
30 05144 151300          MOVS 2,2
31 05145 151232          MOVR 2,2,SZC
32 05146 000432          JMP CK4
33 05147 054462          STA 3,CKRET      ;A ERROR DETECTED
34 05150 044465          STA 1,BAD
35 05151 040465          STA 0,GOOD
36 05152 010460          ISZ FSTGC
37 05153 000425          JMP CK1          ;DON'T PRINT HEADER
38 05154 006351          HEADER           ;"ADDR GOOD BAD WORD
39 05155 006322          PCRLF            ;ETC.
40 05156 006321          MESSAGE
41 05157 005762          MSG22
42 05160 030452 CK1:     LDA 2,FSTGC
43 05161 176120          ADCZL 3,3
44 05162 157036          ADDZ# 2,3,SEZ
45 05163 002446          JMP #CKRET       ;EXIT IF>3 ERRORS.
```

```
10087 .MAIN
01
02 05164 006322          PCRLF
03 05165 024020          LDA 1,IDX0
04 05166 006320          TYPZ1              ;PRINT ADDRESS
05 05167 024447          LDA 1,GOOD
06 05170 006317          TYPAC1             ;GOOD DATA
07 05171 024444          LDA 1,BAD
08 05172 006317          TYPAC1             ;BAD DATA
09
10 05173 020254          LDA 0,CA
11 05174 024020          LDA 1,IDX0
12 05175 106400          SUB 0,1
13 05176 006320          TYPZ1              ;WORD#
14 05177 002432          JMP @CKRET
15
16 05200 010432 CK4:     ISZ FSTGC          ;COUNT ERRORS
17 05201 000401          JMP .+1
18 05202 001400          JMP 0,3            ;RETURN
19
20 05203 006430 CC:      JSR @PATT          ;CHECK THE DATA
21 05204 026020          LDA 1,@IDX0
22 05205 106414          SUB# 0,1,SZR
23 05206 004735          JSR CKER           ;CHECK ERROR
24 05207 010425          ISZ GWC
25 05210 000773          JMP CC
26 05211 010421          ISZ FSTGC          ;ANY ERRORS ??
27 05212 000402          JMP .+2
28 05213 000413          JMP CC1            ;NO, EXIT
29 05214 050477          READS 0            ;IF SW7=1 DON'T PRINT ER ORS
30 05215 101300          MOVS 0,0
31 05216 101202          MOVR 0,0,SZC
32 05217 002411          JMP @GRET
33 05220 006322          PCRLF              ;YES, PRINT THE
34 05221 024411          LDA 1,FSTGC        ;TOTAL # OF ERRORS
35 05222 006320          TYPZ1
36 05223 006321          MESSAGE
37 05224 006166          MSG56              ;"ERRORS"
38 05225 002403          JMP @GRET
39
40 05226 010402 CC1:     ISZ GRET           ;NORMAL RETURN + 2
41 05227 002401          JMP @GRET
42
43 05230 000000 GRET:    0
44 05231 000000 CKRET:   0
45 05232 000000 FSTGC:   0
46 05233 000000 PATT:    0
47 05234 000000 GWC:     0
48 05235 000000 BAD:     0
49 05236 000000 GOOD:    0
```

```
10088 .MAIN
01
02                       ;OPERATOR SPECIFIED WORDS
03
04 05237 054413 VAR,0:   STA 3,VARET
05 05240 010263          ISZ VARPT
06 05241 034263          LDA 2,VARPT
07 05242 034262          LDA 3,VARSP
08 05243 172215          ADC# 3,2,SNR
09 05244 000403          JMP VAR,1          ;END INPUT
10 05245 021000          LDA 0,0,2
11 05246 002404          JMP @VARET
12 05247 030260 VAR,1:   LDA 2,VARST
13 05250 050263          STA 2,VARPT
14 05251 000767          JMP VAR,0+1
15
16 05252 000000 VARET:   0
17
18
19
20                       ;RANDOM NUMBER GENERATOR
21
22 05253 054431 RAN:     STA 3,.UD03        ;GENERATE A RANDOM
23 05254 050427          STA 2,.UD02
24 05255 044425          STA 1,.UD01
25 05256 020244          LDA 0,RANDOM       ;NUMBER IN AC0
26 05257 004410          JSR .UD50
27 05260 034428          LDA 3,.UD20
28 05261 153000          ADD 3,0
29 05262 040244          STA 0,RANDOM       ;STORE NEW VALUE,
30 05263 111100          MOVL 0,2
31 05264 030417          LDA 2,.UD02
32 05265 024415          LDA 1,.UD01
33 05266 002416          JMP @,UD03
34
35 05267 024420 .UD50:   LDA 1,.UD21        ;RANDOM CONTINUED
36 05270 044415          STA 1,.UD10
37 05271 105120          MOVZL 0,1
38 05272 125120          MOVZL 1,1
39 05273 014412          DSZ .UD10
40 05274 000776          JMP .-2
41 05275 107003          ADD 0,1
42 05276 125120          MOVZL 1,1
43 05277 125120          MOVZL 1,1
44 05300 123000          ADD 1,0
45 05301 001400          JMP 0,3
46 05302 000000 .UD01:   0
47 05303 000000 .UD02:   0
48 05304 000000 .UD03:   0
49 05305 000000 .UD10:   0
50 05306 033031 .UD20:   33031
51 05307 000010 .UD21:   10
```

```
10089 .MAIN
01                ; BINARY TO DECIMAL ASCII CONVERT
02                ; CONVERTS A DOUBLE PRECISION, TWO'S COMPLEMENT  UMBER
03                ; TO AN ASCII DECIMAL CHARACTER STRING
04
05                ; INPUT:          D IN AC1, AC2 (HIGH, LOW)
06
07                ; OUTPUT:         ASCII CHARACTER STRING, TERMINAT D BY A
08                ;                 NULL WORD.
09                ;                 CHARACTERS PASSED RIGHT ADJUSTED
10                ;                 BIT 8 = 0, IN AC0 TO TYPING
11                ;                 ROUTINE .
12
13                ;                 STRING OF FORM:
14                ;                         NNNNNNNNNN(NULL)
15
16                ; CALLING SEQUENCE
17                ;       JSR       ,DBD
18                ;       RETURN
19
20                ; DESTROYED:       AC1, AC2, AC3, CARRY
21                ; UNCHANGED:       AC0
22
23 05310 054456 ,DBD:   STA 3,,FD03       ; SAVE RETURN
24 05311 042454          STA 2,,FD00       ; SAVE AC0
25 05312 020502          LDA 0,,FD30       ; POINT TO HIGH ORDER PO ER IN
26                                         ; TABLE
27 05313 040506          STA 0,,FD12
28 05314 176400          SUB 3,3
29 05315 054452          STA 3,DIGIT
30 05316 044500 ,FD99:   STA 1,,FD10       ; SAVE ABS(NUMBER)
31 05317 050500          STA 2,,FD10+1
32 05320 034447          LDA 3,DIGIT
33 05321 175004          MOV 3,3,SZR
34 05322 006502          JSR @BDTYP        ; PUT OUT SIGN OR DIGIT
35 05323 024473          LDA 1,,FD10       ; RESTORE ABS(NUMBER)
36 05324 030473          LDA 2,,FD10+1
37 05325 020476          LDA 0,,FD22       ; GET OCTAL 57
38 05325 040472          STA 0,,FD11       ; COUNT IT UP IN STORAGE
39 05327 034472          LDA 3,,FD12       ; CURRENT POINTER TO POW R OF
40                                         ; 10 TABLE
41
42
43 05330 021421 ,FD98:   LDA 0,1,3         ; LOW ORDER WORD
44 05331 101235          MOVZR# 0,0,SNR
45 05332 010435          ISZ DIGIT
46 05333 101005          MOV 0,0,SNR       ; TEST FOR END OF TABLE
47 05334 000426          JMP .FD97         ; DONE
48 05335 112420          SUBZ 0,2
49 05336 021400          LDA 0,0,3         ; HIGH ORDER WORD
50 05337 101003          MOV 0,0,SNC
51 05340 106041          ADCO 0,1,SKP
52 05341 106440          SUBO 0,1
53 05342 010456          ISZ .FD11         ; COUNT UP DIGIT
54 05343 125013          MOV# 1,1,SNC      ; TEST FOR <0
55 05344 000764          JMP .FD98         ; KEEP SUBTRACTING
56
```

```
10090 .MAIN
01
02 05345 021401          LDA 0,1,3         ; RESTORE POSITIVE VALUE
03 05346 113022          ADDZ 0,2,SZC
04 05347 125400          INC 1,1
05 05350 021400          LDA 0,0,3
06 05351 107000          ADD 2,1
07 05352 175400          INC 3,3           ; BUMP AC3 TO NEXT TABLE ENTRY
08 05353 175400          INC 3,3
09 05354 054445          STA 3,.FD12
10 05355 020443          LDA 0,.FD11       ; GET DIGIT
11 05356 034217          LDA 3,C60
12 05357 116414          SUB# 0,3,SZR
13 05360 010407          ISZ DIGIT
14 05361 000735          JMP .FD99         ; PUT IT OUT
15
16 05362 006321 .FD97:   JSR @IMESS
17 05363 005774          MSG23             ;"TAB"
18
19 05364 002402          JMP @.FD03        ; RETURN
20
21 05365 000000 .FD00:   0                 ; SAVE AC0
22 05366 000000 .FD03:   0                 ; SAVE RETURN
23 05367 000000 DIGIT:   0
24
25 05370 035632 .FD05:   35632             ; 10**9
26 05371 145000          145000
27 05372 002765          2765              ; 10**8
28 05373 150400          150400
29 05374 000230          230               ; 10**7
30 05375 113200          113200
31 05376 000017          17                ; 10**6
32 05377 041100          41100
33 05400 000001          1                 ; 10**5
34 05401 103240          103240
35       000012          .RDX 10
36 05402 000000          0                 ; 10**4
37 05403 023420          10000
38 05404 000000          0                 ; 10**3
39 05405 001750          1000
40 05406 000000          0                 ; 10**2
41 05407 000144          100
42 05410 000000          0                 ; 10**1
43 05411 000012          10
44 05412 000000          0                 ; 10**0
45 05413 000001          1
46 05414 005370 .FD30:   .FD05   ; POINTER TO CONVERSION TABLE
47 05415 000000          0       ; END OF TABLE INDICATION
48       000010          .RDX 8
49
50 05416 000002 .FD10:   .BLK 2            ; SAVE CURRENT DOUBLE WO D
51 05420 000000 .FD11:   0                 ; COUNT UP DIGIT WORD
52 05421 000000 .FD12:   0                 ; POINTER TO POWER OF TE  ENTRY
53
54 05422 000053 .FD20:   "+                ; ASCII "+"
55 05423 000057 .FD22:   57                ; ASCII "0" -1
56 05424 004220 BDTYP:   TYPE              ;PUT CHARACTER ADDRESS.
```

```
;0091 .MAIN
01
02                              ;SAVE AC0,1,2
03 05425 040240 SAC:    STA 0,SAV0
04 05426 044241         STA 1,SAV1
05 05427 050242         STA 2,SAV2
06 05430 001400         JMP 0,3
07
08                              ;RESTORE AC0,1,2
09 05431 020240 STAC:   LDA 0,SAV0
10 05432 024241         LDA 1,SAV1
11 05433 030242         LDA 2,SAV2
12 05434 001400         JMP 0,3
13
```

```
;0092 .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 05435 054514 INP:    STA 3,INPRET
13 05436 023400         LDA 0,@0,3      ;GET BYTE POINTER
14 05437 040511         STA 0,BASE
15 05440 040512         STA 0,BPTR
16 05441 060210         NIOC TTI
17 05442 020512         LDA 0,M115,
18 05443 040510         STA 0,CHCNT
19 05444 063610 TTWAIT: SKPDN TTI       ;WAIT FOR INPUT
20 05445 000777         JMP .-1
21 05446 064610         DIAC 1,TTI      ;READ INPUT CHAR
22 05447 020225         LDA 0,C377
23 05450 106415         SUBM 0,1,SNR
24 05451 000442         JMP RUB         ;RUB OUT
25 05452 020503         LDA 0,C12
26 05453 106414         SUBM 0,1,SZR
27 05454 000403         JMP .+3
28 05455 006322         PCRLF           ;LINE FEED TYPED
29 05456 000766         JMP TTWAIT
30 05457 065111         DOAS 1,TTO      ;ECHO CHAR
31 05460 030472         LDA 2,BPTR
32 05461 010471         ISZ BPTR
33 05462 034224         LDA 3,C215      ;CR CODE
34 05463 136405         SUB 1,3,SNR
35 05464 000456         JMP CRCOD       ;CR TYPED
36 05465 020471         LDA 0,C177      ;7 BIT MASK
37 05466 151223 INP,0:  MOVZR 2,2,SNC
38 05467 107401         AND 0,1,SKP     ;STORE INTO RH
39 05470 107701         ANDS 0,1,SKP
40 05471 000403         JMP .+3
41 05472 021000         LDA 2,0,2
42 05473 107000         ADD 0,1
43 05474 045000         STA 1,0,2       ;STORE BYTE
44 05475 010456         ISZ CHCNT
45 05476 000402         JMP .+2
46 05477 000435         JMP OVFL
47 05500 175004         MOV 3,3,SZR     ;AC3=0 IF CR TYPED
48 05501 000743         JMP TTWAIT
49 05502 010447         ISZ INPRET      ;ALL DONE
50 05503 006322 INP,1:  PCRLF
51 05504 020444         LDA 0,BASE
52 05505 040250         STA 0,LINCT
53 05506 010443         ISZ INPRET
54 05507 063511         SKPBZ TTO       ;WAIT FOR ECHO TO
55 05510 000777         JMP .-1         ;FINISH
56 05511 060211         NIOC TTO        ;CLEAR INTERRUPT
57 05512 002437         JMP @INPRET
```

```
10093 .MAIN
01
02 05513 020441 RUB:    LDA 0,M115.
03 05514 024437         LDA 1,CHCNT
04 05515 122425         SUB 1,0,SNR
05 05516 000726         JMP TTWAIT      ;NOTHING TO RUB
06 05517 014433         DSZ BPTR
07 05520 014433         DSZ CHCNT
08 05521 030431         LDA 2,BPTR
09 05522 151220         MOVZR 2,2
10 05523 021000         LDA 0,0,2       ;GET BYTE
11 05524 101003         MOV 0,0,SNC
12 05525 000405         JMP RUB1
13 05526 024225         LDA 1,C377
14 05527 107400         AND 0,1         ;SAVE RH ONLY
15 05530 045000         STA 1,0,2
16 05531 101300         MOVS 0,0        ;ECHO LH (RUBBED OUT)
17 05532 061111 RUB1:   DOAS 0,TTO      ;ECHO RUBBED CHAR
18 05533 000711         JMP TTWAIT
19
20 05534 006322 OVFL:   PCRLF           ;LINE OVERFLOW
21 05535 006321         MESSAGE
22 05536 005752         MSG21           ;"INPUT OVERFLOW"
23 05537 014413         DSZ BPTR        ;BACKUP BYTE POINTER
24 05540 014413         DSZ CHCNT
25 05541 000703         JMP TTWAIT
26
27 05542 020406 CRCOD:  LDA 0,BASE
28 05543 112415         SUBA 0,2,SNR
29 05544 000737         JMP INP,1       ;YES
30 05545 020225         LDA 0,C377
31 05546 176400         SUB 3,3
32 05547 000717         JMP INP,0
33
34 05550 000000 BASE:   0
35 05551 000000 INPRET: 0
36 05552 000000 BPTR:   0
37 05553 000000 CHCNT:  0
38 05554 177615 M115.:  -115.
39 05555 000012 C12:    12
40 05556 000177 C177:   177
```

```
10094 .MAIN
01                      ;DISK COMMAND TABLE
02
03 05557 044241 DCT:    044241          ;READ
04 05560 057111         057111          ;WRITE
05 05561 046245         046245          ;SEEK
06 05562 044243         044243          ;RECALIBRATE
07 05563 030757         030757          ;LOOP
08 05564 010254         010254          ;DELAY
09 05565 000000         0
10
11                      ;DISPATCHES
12
13 05566 002723 DCT.1:  RE
14 05567 002737         WT
15 05570 002752         SK
16 05571 002771         RCL
17 05572 003002         LUP
18 05573 003007         DLAY
19
20                      ;DATA TABLE
21
22 05574 014632 DNT:    14632           ;FLOATING ZERO
23 05575 014617         14617           ;FLOATING ONE
24 05576 044056         044056          ;RANDOM
25 05577 002617         2617            ;ALL 1'S
26 05600 002632         2632            ;ALL ZEROS
27 05601 040064         40064           ;110110 PAT
28 05602 000000         0
29
30                      ;DISPATCHES
31
32 05603 005626 DNT.1:  FLZ
33 05604 005620         FL1
34 05605 005253         RAN
35 05606 005612         ONES
36 05607 005611         ZEROS
37 05610 005615         PAT1
```

```
↓0095 .MAIN
01
02 05611 102401 ZEROS:  SUB 0,0,SKP
03 05612 102000 ONES:   ADC 0,0
04 05613 001400         JMP 0,3
05
06 05614 155555         155555
07 05615 020777 PAT1:   LDA 0,,-1
08 05616 001400         JMP 0,3
09
10 05617 000000         0
11 05620 020777 FL1:    LDA 0,,-1
12 05621 101225         MOVZR 0,0,SNR
13 05622 101240         MOVOR 0,0
14 05623 040774         STA 0,FL1-1
15 05624 001400         JMP 0,3
16
17 05625 000000         0
18 05626 020777 FLZ:    LDA 0,,-1
19 05627 101243         MOVOR 0,0,SNC
20 05630 102220         ADCZR 0,0
21 05631 040774         STA 0,FLZ-1
22 05632 001400         JMP 0,3
23
24 05633 020362 CYLADD: LDA 0,CYLNX
25 05634 001400         JMP 0,3
26
```

```
↓0096 .MAIN
01
02                      ↓ENDING STATUS
03 05635 047305 MSG1:   .TXTE ↓ENDING STATUS↓
04       144504
05       043516
06       051640
07       040724
08       052724
09       000123
10                      ↓RECAL
11 05644 142722 MSG2:   .TXTE ↓RECAL     ↓
12       040703
13       004714
14       000000
15                      ↓SEEK
16 05650 142523 MSG3:   .TXTE ↓SEEK      ↓
17       045705
18       000011
19                      ↓INTERRUPT FROM DEVICE
20 05653 047311 MSG9:   .TXTE ↓INTERRUPT FROM DEVICE ↓
21       142724
22       151322
23       050125
24       120324
25       151305
26       045717
27       042240
28       053305
29       141711
30       120305
31       000000
32                      ↓DISK
33 05667 144504 MSG13:  .TXTE ↓DISK                      ↓
34       045523
35       004411
36       000011
37                      ↓SEEK ERRORS
38 05673 142523 MSG14:  .TXTE ↓SEEK ERRORS               ↓
39       045705
40       142640
41       151322
42       151317
43       004523
44       000011
45                      ↓TOTAL SEEKS
46 05702 147724 MSG15:  .TXTE ↓TOTAL SEEKS               ↓
47       040724
48       120314
49       142523
50       045705
51       004523
52       000011
53                      ↓NO READY UNITS
54 05711 147516 MSG16:  .TXTE ↓NO READY UNITS↓
55       151240
56       040705
57       054504
58       052640
59       144516
60       051724
```

```
 0097 .MAIN
01        000000
02                            /UNIT:
03 05721 047125 MSG17:  .TXTE |UNIT: |
04        152311
05        122072
06        000000
07                            /DATA:
08 05725 040504 MSG18:  .TXTE |DATA: |
09        040724
10        122072
11        000000
12                            /COMMAND STRING:
13 05731 147703 MSG19:  .TXTE |COMMAND STRING: |
14        046515
15        047101
16        120104
17        152123
18        144722
19        043516
20        122072
21        000000
22                            /16 NUMBERS MAX.
23 05742 033261 MSG20:  .TXTE |16 NUMBERS MAX.|
24        047240
25        046525
26        142522
27        051722
28        046640
29        154101
30        000056
31                            /INPUT OVERFLOW
32 05752 047311 MSG21:  .TXTE |INPUT OVERFLOW|
33        052520
34        120324
35        053317
36        151305
37        146306
38        153717
39        000000
40                            /ADDR   GOOD    BAD     WORD
41 05762 042101 MSG22:  .TXTE |ADDR    GOOD    BAD     WORD|
42        151104
43        043411
44        147717
45        004504
46        040522
47        004504
48        147727
49        042322
50        000020
51                            / TAB
52 05774 000011 MSG23:  .TXTE | |
53                            /WORDS WRITTEN
54 05775 147727 MSG25:  .TXTE |WORDS WRITTEN           |
55        042322
56        120123
57        151327
58        152311
59        142724
60        004516
```

```
 0098 .MAIN
01        000011
02                            /WORDS READ
03 06005 147727 MSG26:  .TXTE |WORDS READ              |
04        042322
05        120123
06        142722
07        042101
08        004411
09        000000
10                            /CHECK WORD ERRORS
11 06014 044303 MSG27:  .TXTE |CHECK WORD ERRORS       |
12        141705
13        120113
14        147727
15        042322
16        142640
17        151322
18        151317
19        004523
20        000000
21                            /PERM CHECK WORD ERRORS
22 06026 142520 MSG28:  .TXTE |PERM CHECK WORD ERRORS  |
23        046722
24        141640
25        142510
26        045703
27        153640
28        151317
29        120104
30        151305
31        147722
32        051722
33        000011
34                            /ADDRESS ERROR
35 06042 042101 MSG29:  .TXTE |ADDRESS ERROR           |
36        151104
37        051705
38        120123
39        151305
40        147722
41        004722
42        000011
43                            /PERM ADDRESS ERROR
44 06052 142520 MSG30:  .TXTE |PERM ADDRESS ERROR      |
45        046722
46        040640
47        042104
48        142722
49        051523
50        142640
51        151322
52        151317
53        000011
54                            /DATA ERRORS
55 06064 040504 MSG31:  .TXTE |DATA ERRORS            |
56        040724
57        142640
58        151322
59        151317
60        004523
```

```
 0099 .MAIN
01        000011
02                            ;READ
03 06073 142722 MSG32:  .TXTE |READ     |
04        042101
05        000011
06                            ;WRITE
07 06076 151327 MSG33:  .TXTE |WRITE    |
08        152311
09        004705
10        000000
11                            ;MEMORY TOO SMALL
12 06102 142515 MSG39:  .TXTE |MEMORY TOO SMALL FOR |
13        147515
14        054722
15        152240
16        147717
17        051640
18        040515
19        146314
20        143240
21        151317
22        000240
23                            ;SECTORS
24 06115 142523 MSG40:  .TXTE |SECTORS|
25        152303
26        151317
27        000123
28                            ;PASS
29 06121 040520 MSG41:  .TXTE |PASS|
30        051523
31        000000
32                            ; -?-
33 06124 026640 MSG43:  .TXTE | -?-|
34        026477
35        000000
36                            ;CYL-
37 06127 054703 MSG44:  .TXTE |CYL-|
38        026714
39        000000
40                            ;HEAD-
41 06132 142510 MSG45:  .TXTE |HEAD-|
42        042101
43        000055
44                            ;SECT-
45 06135 142523 MSG46:  .TXTE |SECT-|
46        152303
47        000055
48                            ;#SECT-
49 06140 051543 MSG47:  .TXTE |#SECT-|
50        141705
51        026724
52        000000
53                            ;GOOD
54 06144 147507 MSG51:  .TXTE |GOOD |
55        042317
56        002240
57                            ;BAD
58 06147 040502 MSG52:  .TXTE |BAD |
59        120104
60        000000
```

```
 0100 .MAIN
01                            ;"DOC"
02 06152 042042 MSG53:  .TXTE |"DOC"|
03        141717
04        000042
05                            ;"DOB"
06 06155 042042 MSG54:  .TXTE |"DOB"|
07        041317
08        000042
09                            ;LOAD ERROR
10 06162 146240 MSG55:  .TXTE | LOAD ERROR|
11        040717
12        120104
13        151305
14        147722
15        000322
16                            ;ERRORS
17 06166 151305 MSG56:  .TXTE |ERRORS|
18        147722
19        051722
20        000000
21                            ;ENDING MEM ADDR ERROR
22 06172 047305 MSG57:  .TXTE |ENDING MEM ADDR ERROR|
23        144504
24        043516
25        046640
26        046705
27        040640
28        042104
29        120322
30        151305
31        147722
32        000322
33                            ;TIMEOUT
34 06205 144724 MSG59:  .TXTE |TIMEOUT|
35        142515
36        052717
37        000324
38                            ;PERM DATA ERRORS
39 06211 142520 MSG61:  .TXTE |PERM DATA ERRORS |
40        046722
41        042240
42        152101
43        120121
44        151305
45        147722
46        051722
47        000011
48                            ;MISC ERRORS
49 06222 144515 MSG62:  .TXTE |MISC ERRORS              |
50        141523
51        142640
52        151322
53        151317
54        004523
55        000011
56                            ;TRY AGAIN
57 06231 151324 MSG63:  .TXTE |TRY AGAIN|
58        120131
59        043501
60        144501
```

```
 0101 ,MAIN
01        000116
02                          ;CHECK WORD ERROR
03 06236 044303 MSG64:      ,TXTE |CHECK WORD ERROR|
04        141705
05        120113
06        147727
07        042322
08        142640
09        151322
10        151317
11        000000
12                          ;CHECK WORD & DATA ERROR
13 06247 044303 MSG65:      ,TXTE |CHECK WORD/DATA ERRORS    |
14        141705
15        120113
16        147727
17        042322
18        042257
19        152101
20        120101
21        151305
22        147722
23        051722
24        000011
25                          ;PERM CHK WD/DATA ERRS
26 06283 142520 MSG66:      ,TXTE |PERM CHK WD/DATA ERRS     |
27        046722
28        141640
29        045510
30        153640
31        127504
32        040504
33        040724
34        142640
35        151322
36        004523
37        000000
38                          ;ENDING DISK ADDR ERROR
39 06277 047305 MSG67:      ,TXTE |ENDING DISK ADDR ERROR|
40        144504
41        043516
42        042240
43        051711
44        120113
45        042101
46        151104
47        142640
48        151322
49        151317
50        000000
51                          ;INTERCHANGE DISK
52 06313 047311 MSG68:      ,TXTE |INTERCHANGE DISK|
53        142724
54        141722
55        040510
56        043516
57        120305
58        144504
59        045523
60        000000
```

```
 0102 ,MAIN
01                          ;TYPE THE NUMBER OF DISK SURFACES
02 06324 054724 MSG69:      ,TXTE |TYPE THE NUMBER OF DISK SURFACES
03        142520
04        152240
05        142510
06        047240
07        046525
08        142502
09        120322
10        143317
11        042240
12        051711
13        120113
14        052523
15        143322
16        141501
17        051705
18        000240
19                          ;TESTING UNIT
20 06345 142724 MSG70:      ,TXTE |TESTING UNIT  |
21        152123
22        047311
23        120107
24        047125
25        152311
26        120240
27        000000
28 06355 142523 MSG71:      ,TXTE [SEEK ERROR PATH BETWEEN    [
29        045705
30        142640
31        151322
32        151317
33        050240
34        152101
35        120110
36        142502
37        153724
38        142705
39        120116
40        120240
41        000240
42 06373 142523 MSG72:      ,TXTE [SEEK/RECAL TIMEOUTS      [
43        045705
44        151257
45        141705
46        146101
47        152240
48        045711
49        147705
50        152125
51        004523
52        000000
53 06406 142722 MSG73:      ,TXTE [READ/WRITE TIMEOUTS      [
54        042171
55        153657
56        144722
57        142724
58        152240
59        046711
60        147735
```

```
 0103 .MAIN
01        152125
02        004523
03        000000
04 06421  151305 MSG76:  .TXTE [ERROR STATUS AFTER WRITE =
05        147722
06        120322
07        152123
08        152101
09        051525
10        042040
11        152305
12        151305
13        153640
14        144722
15        142724
16        136640
17        000011
18 06437  142504 MSG77:  .TXTE [DEVICE CODE=]
19        144526
20        142703
21        141640
22        042317
23        136705
24        000000
25 06446  005215 MSG78:  .TXTE ]<15><12>MOUNT SCRATCH PACK(S),SET PROGRAM SWITCHE
26        147515
27        047125
28        120324
29        141523
30        040722
31        141724
32        120110
33        040520
34        045703
35        051450
36        125251
37        142523
38        120324
39        151120
40        043717
41        040722
42        120115
43        153523
44        152311
45        044303
46        051705
47 06474  005215 <15><12>THEN PRESS CONTINUE KEY<15><12>]
48        044324
49        047305
50        050240
51        142722
52        051523
53        141640
54        047317
55        144724
56        052516
57        120305
58        142513
59        106531
60        000012
```

```
 0104 .MAIN
01 06512  151120 MSGD1:  .TXTE [PRESENT DEVICE CODE      [
02        051705
03        047305
04        120324
05        142504
06        144526
07        142703
08        141640
09        042317
10        120305
11        120240
12        122240
13        000220
14 06527  054724 MSGD2:  .TXTE [TYPE DEVICE CODE W/CR OR JUST CR  F PRESENT
15        142520
16        042240
17        053305
18        141711
19        120305
20        147703
21        142504
22        153640
23        141657
24        120322
25        151317
26        145240
27        051525
28        120324
29        151303
30        144640
31        120306
32        151120
33        051705
34        047305
35 06554  106724 <15><12>CODE IS OK, [
36        141412
37        042317
38        120305
39        051711
40        147640
41        027113
42        020240
43 06564  142504 MSGD3:  .TXTE [DEVICE CODE SET TO       [
44        144526
45        142703
46        141640
47        042317
48        120305
49        142523
50        122324
51        147724
52        000011
53 06576  142722 MSGD4:  .TXTE [RESTART PROGRAM AFTER HALT, [
54        152123
55        151101
56        120324
57        151120
58        043717
59        040722
60        120115
```

```
 0105 .MAIN
01        143171
02        142724
03        120322
04        040510
05        152314
06        120056
07        000000
```

```
10106 .MAIN
01
02
03
04 06615 000272 UBUFF:   .BLK 58.
05 06727 000272 CBUFF:   .BLK 58.
06 07001 000020 VAR:     .BLK 16.
07
08 07021 000000 PRGEND: 0
09 07022 047503          .TXT    /COPYRIGHT (C) DGC, 1971,72,73,7 ,75,76
10        054520
11        044522
12        044107
13        020124
14        041450
15        020051
16        043504
17        026103
18        030440
19        033471
20        026061
21        031067
22        033454
23        026063
24        032067
25        033454
26        026065
27        033067
28 07045 046101 ALL RIGHTS RESERVED,/
29        020114
30        044522
31        044107
32        051524
33        051040
34        051505
35        051105
36        042526
37        027104
38        000000
39
40 07060 045504 DIRT:    .TXTE   !DKP RELI 091
41        122120
42        142722
43        144714
44        030240
45        000071
46 07066 000001          000001
47 07067 000270          PROGB
48 07070 000001          000001
49 07071 070707          70707
50 07072 070707          70707
51 07073 070707          70707
52 07074 070707          70707
53 07075 100033          100033
54
55                       .END

**00000 TOTAL ERRORS, 00000 PASS 1 ERRORS
```

```
                26/49   27/13   27/17   29/35   30/06   31/28   35/10
                35/18   36/41   37/23   39/37   47/25   49/08   49/15
                50/24   50/07   51/19   51/33   51/48   58/07   58/17
                58/35   60/25   60/28   60/32   60/44   65/04   65/09
                65/28   65/33   85/14   85/16   85/23   85/27   85/31
                85/35   86/40   87/36   93/21
MESSR 004217    68/26   68/29   68/50   69/09   70/23   70/33   72/03
                72/12
MISC  000156    12/20   27/05   28/43   57/41   57/48   58/02   60/05
                60/12   60/41
MLOOP 004032    67/24   67/28
MODE  000204    12/37   25/11   54/12   55/08   56/11   61/12   85/08
MS1   001501    30/20   31/18
MS2   001505    30/22   31/23
MSAV  004041    67/04   67/27   67/18   67/22   67/30   67/31
MSG1  005635    29/36   96/03
MSG13 005667    27/25   96/33
MSG14 005673    27/29   96/38
MSG15 005702    27/27   96/46
MSG16 005711    22/48   96/54
MSG17 005721    49/16   85/36   97/03
MSG18 005725    50/28   97/08
MSG19 005731    51/34   97/13
MSG2  005644    85/42   96/11
MSG20 005742    51/20   97/23
MSG21 005752    93/22   97/32
MSG22 005762    86/41   97/41
MSG23 005774    47/26   92/17   97/52
MSG25 005775    27/31   97/54
MSG26 006005    27/33   98/03
MSG27 006014    27/39   98/11
MSG28 006026    27/41   98/22
MSG29 006042    27/35   58/08   98/35
MSG3  005650    85/43   96/16
MSG30 006052    27/37   98/44
MSG31 006064    27/43   98/55
MSG32 006073    85/44   99/03
MSG33 006076    85/45   99/27
MSG39 006102    27/14   99/12
MSG40 006115    27/18   99/24
MSG41 006121    35/11   35/19   36/42   37/24   99/29
MSG43 006124    31/29   49/09   50/05   51/49   99/33
MSG44 006127    85/17   99/37
MSG45 006132    85/24   99/41
MSG46 006135    85/28   99/45
MSG47 006142    85/32   99/49
MSG51 006144    26/46   60/29   99/54
MSG52 006147    26/50   60/33   99/58
MSG53 006152    26/30   100/02
MSG54 006155    26/41   100/06
MSG55 006160    26/43   100/10
MSG56 006166    87/37   100/17
MSG57 006172    57/39   60/03   100/22
MSG59 006205    24/30   100/34
MSG61 006211    27/45   100/39
MSG62 006222    27/51   100/49
MSG63 006231    58/16   100/57
MSG64 006236    58/36   101/03
```



```
MSG65 006247    27/47   101/13
MSG66 006263    27/49   101/26
MSG67 006277    57/46   60/10   101/39
MSG68 006313    39/38   101/52
MSG69 006324    39/07   102/02
MSG70 006345    18/35   102/20
MSG71 006355    47/22   102/28
MSG72 006373    27/53   102/42
MSG73 006406    27/55   102/53
MSG76 006421    60/45   103/04
MSG77 006437    18/11   19/18   103/18
MSG78 006445    18/16   19/23   103/25
MSGS  005653    23/22   96/20
MSGD1 006512    65/05   104/01
MSGD2 006527    65/10   104/14
MSGD3 006564    65/29   104/43
MSGD4 006575    65/34   104/53
MSKBS 001056    24/12
MULT  004027    67/21
MULTA 004030    67/22   78/10
NEST  000045    11/36
NOP   000401    16/34   53/29   55/19   57/26   82/38
NO.BS 000353    15/44   56/18   61/20
NUCAL 004530    74/35   76/17   76/22   76/28   77/45   78/05
NUCCO 000371    11/10   15/50
OCTAB 004134    69/04   69/34
OMEGA 000235    14/02   43/21   48/08   48/11   48/19   48/23   81/05
                81/15
ONES  005612    36/28   94/35   95/03
ORDIN 004527    75/22   75/40   75/41   76/30   77/45   78/04
OVFL  005534    92/46   93/20
PADD  001756    36/13   36/14   35/15   36/23
PADER 002132    12/15   28/34   58/26
PAT1  005615    36/29   94/37   95/07
PATT  005233    86/09   86/22   87/20   87/46
PCRLF 005322    16/13   18/09   18/14   18/33   19/16   19/21   22/46
                23/20   24/28   25/14   25/15   25/28   26/28   26/39
                26/44   27/12   29/32   30/05   35/09   35/17   36/40
                37/22   39/36   49/14   50/06   51/18   51/32   58/06
                58/16   58/34   60/24   60/27   60/43   65/03   65/08
                65/27   65/32   65/35   85/12   85/13   86/39   87/02
                87/33   92/28   92/50   93/20
PDEC  004101    15/07   69/06
PDER  002152    12/19   28/40   59/27
PEX   004351    72/08   72/57
PEXIT 004063    68/42   72/57
PHDR  001325    27/26   28/01
POCT  004075    15/08   69/02
POT   001264    25/20   27/25
PRGEN 007021    14/12   14/17   14/25   106/08
PROGB 000200    12/33   21/41   106/47
PRNTO 001374    27/56   28/55
PS    001421    15/33   29/31
PSKER 001331    27/30   28/06
PSKTO 001372    27/54   28/52
PSTAT 002350    16/25   54/31   54/36   55/40   58/03   60/50
PS.1  001332    26/27   28/26   28/29   28/32   28/35   28/38   28/41
                28/44   28/47   28/50   28/53   28/56
```

```
PTAER 001354    27/36   28/31
PTCND 001366    27/48   28/46
PTCNE 001350    27/40   28/25
PTCNP 001352    27/42   28/28
PTDER 001360    27/44   28/37
PTIME 004352    13/12   74/16
PTMS  001364    27/52   28/43
PTPAE 001356    27/38   28/34
PTPCN 001370    27/50   28/49
PTPDE 001362    27/46   28/40
PTSEK 001335    27/28   28/11
PTWR  001345    27/34   28/21
PTWW  001340    27/32   28/15
PT,1  001342    28/13   28/17   28/23
P,240 004257    70/14   71/32
P,377 004160    68/31   69/56   71/10
P,AC1 004154    68/27   68/42   69/10   69/52   70/24   72/04   72/21
                72/46
P,AC2 004155    68/28   68/43   69/03   69/06   69/53   70/25   72/13
P,C11 004254    70/09   71/29
P,C12 004255    70/28   71/30   72/01   72/14
P,C15 004256    70/26   71/17   71/31
P,C40 004260    71/14   71/33   72/26
P,C60 004157    69/02   69/23   69/27   69/55
P,C7  004253    70/17   71/28
P,LST 004062    68/41   69/17   70/29
P,TAB 004156    69/15   69/54
QUEST 002572    15/38   49/08   49/24   49/27   49/30
RADD  001757    36/17   36/24
RALL  000177    12/29   21/37   35/03   36/11
RAN   005253    15/40   36/32   88/22   94/34
RAND  006357    16/06   32/20   32/29   32/37   32/45   33/21
RANDO 000244    14/09   21/43   34/08   34/11   34/32   37/03   38/25
                38/29   39/04   39/07   39/34   52/08   52/21   52/24
                58/20   88/25   88/29
RATIO 001635    32/17   33/38   34/43   34/46
RCL   002771    53/22   94/16
RCTYM 003057    54/22   54/38
RDATA 003525    15/26   61/46
RDCYL 002450    43/34   44/05   45/19   45/22   45/25   45/28   45/33
                46/22
RDO   000201    12/34
RDYUN 000175    12/27   22/23   22/43   29/12   32/24   49/21
RE    002723    52/04   94/13
RE1   002733    52/12   52/28   53/18   53/30
READ  005344    16/26   34/33   39/43   46/26   52/09
RECAL 005347    16/29   38/35   40/25   47/33   53/24
RECL  003023    15/32   54/10
RELAL 001517    19/38   32/09   36/24
RELEV 001515    19/39   32/07
RELOD 001514    19/40   32/06
RELRA 000245    14/10   34/09   34/10   34/31   39/05   39/06   52/07
                52/22   52/23   58/19
REL,1 001530    32/20   32/26   35/08   35/22
REL,2 001540    32/29   32/34   32/42
REL,3 001547    32/37
REL,4 001556    32/45   32/51   33/19
REL,5 001572    33/02   33/09   33/16
```



```
REL,6 001614    33/04   33/21   33/32
REL,7 001625    33/26   33/30
REL,8 001714    34/42   34/44   35/03
REL,9 001730    35/06   35/15
REST  000744    21/40   21/48
RETRY 000202    12/35   58/22   58/41   59/06   61/18   62/43
RE,1  003252    58/02   62/50
RE,2  003264    56/43   58/25   62/51
RE,20 003274    58/13   62/51   62/47
RE,21 003310    58/11   58/26
RE,3  003312    58/29   62/54
RE,31 003330    58/39   58/43
RE,32 003332    58/42   58/45
RE,4  003343    59/02   61/47
RE,41 003354    59/08   59/11
RE,42 003361    59/17   59/23
RE,43 003363    59/13   59/19
RE,44 003370    59/05   59/24
RE,45 003375    59/26   59/29
RE,46 003400    59/28   59/32
RE,5  003402    60/02   61/55
RE,6  003410    60/09   62/30
RFLG  002153    38/19   39/21   39/42   40/14   41/18
RLUP  002306    14/43   56/09   58/24   61/10
RS3   000544    18/25   18/29
RSRD  003476    61/07   61/21
RSTRT 000531    18/18
RUB   005513    92/24   93/02
RUB1  005532    93/12   93/17
RUNAL 001740    19/49   35/20   36/09
RVTMP 004526    77/09   77/47   78/03
RWRET 002273    14/32   29/37   54/10   54/28   55/06   55/25   56/07
                56/34   57/30   57/33   57/34   57/35   57/36   58/15
                58/27   61/08   61/31   62/42
RWTMO 000166    12/22   28/55   58/33   61/30
R,1   001745    36/14   36/21
S0    001050    23/09   23/23   23/29
SAC   005425    15/05   91/03
SAM   003654    63/11   63/39
SAV0  000240    14/05   27/02   27/15   32/56   33/02   33/17   60/34
                91/03   91/09
SAV1  000241    14/06   26/47   33/05   33/06   33/11   60/30   91/04
                91/10
SAV2  000242    14/07   33/12   33/13   91/05   91/11
SAV3  000243    14/08   23/06   23/26   25/41
SAVA  001150    25/19   25/38   25/45
SAVAC 006314    16/07   23/07   26/27   26/38   27/11   32/55   60/22
SAVU  001147    25/17   25/36   25/44
SC    P00252    14/15   25/51   26/06   32/52   38/58   39/18   40/50
                41/15   43/18   56/19   56/45   57/02   57/21   62/06
                62/22   63/34   85/33
SCANX 002432    43/10   45/47
SCNT  002261    15/36   39/14   39/17   41/11   41/14   42/10
SCORA 004377    75/09   75/17
SCORE 004372    74/18   74/23   75/04
SCRET 002311    42/17   42/23   42/35   42/36
SC,1  002266    42/16   42/21
SC,2  002272    42/20   42/29
```

```
$C,3  002276      42/19   42/25
SEARC 006323      16/19   32/13   50/15   51/43
SEC   000251      14/14   26/16   32/43   32/53   38/55   39/15   40/47
                  41/12   42/13   63/28   85/29
SEEK  006346      16/28   33/47   34/21   38/49   40/42   46/11   53/15
SEEKT 000062      12/08   28/11   28/12   55/16   55/18
SEKER 000072      12/09   28/26   55/39
SEKET 001250      26/04   26/37   27/08   27/09
SESCU 004542      75/10   78/15
SET   001163      15/20   26/04   27/07
SET1  001217      26/26   26/33
SET2  001230      26/31   26/42
SETAC 006315      16/08   23/25   25/40
SETB  006335      16/23   18/19   19/26
SETP  006333      16/21   56/23   61/22
SETU  006361      16/36   18/22
SETU, 000745      22/04   22/25
SE,1  003116      55/29   55/34   55/37
SK    002752      53/04   94/15
SKIP  101011      16/21   54/24   55/21   56/30   61/27
SKTMO 000162      12/21   28/52   54/27   55/24
SLNCT 005434      83/14   84/16   84/27
SM    004664      15/21   82/22
SMAX  000266      14/27   26/28   32/49   38/57   40/49   82/43   82/55
SMEM  005334      16/22   18/18   19/25
SMSK  000303      14/40   32/23   32/38
SMX   001475      30/15   31/13
SM,1  004667      82/25   82/33
SM,2  004700      82/30   82/34
SM,3  004716      82/48   82/56
SM,4  004721      82/40   82/53
SM,5  004725      82/22   82/50   82/57
SPRSK 002313      41/23   43/10
SQIK1 002565      48/35   48/36   49/33
SRCH  005035      83/15   83/22   83/41   84/28
SRH   004653      15/12   82/09   82/16
SSW   000206      12/39   21/31
ST    000044      11/12   11/35   12/33   21/48
STAC  005431      15/06   91/09
STB0  000675      15/22   21/05
STB1  000714      21/14   21/18   21/20
STB2  000716      21/11   21/19   21/22
STC   002337      43/31   44/08   44/16   45/10
STC,1 002343      44/02
STC,2 002362      44/13   45/02
STC,3 002373      45/04   45/15   45/30
STC,A 002360      44/15
STIME 000221      13/12   21/12
STRC  000601      17/12   19/16
STRT  000515      17/07   17/08   17/09   17/10   17/13   17/14   17/15
                  17/16   17/17   17/18   17/19   18/06   20/07   65/25
STR,2 000620      18/44   19/37
SUBSE 002434      43/20   43/32   44/23   45/18   45/21   45/24   45/27
                  45/32   46/04
SVLHD 000365      15/45   40/09   45/34
SVSTB 000721      21/05   21/20   21/22   21/26
SVTIM 004531      74/15   74/29   76/31   78/06
SWCE  003416      57/38   57/45   60/02   60/09   60/19
```



```
SWCE1 003425      60/21   60/26
SWCRE 003441      60/19   60/36   60/37   62/38
S,1   001251      26/10   27/11
S,3D1 004541      75/19   75/38   76/25   78/14
TBIG  002427      43/28   45/20   45/44   48/15
TERM  000255      14/18   51/05   52/12   53/08   53/23   63/36
TIMA  004474      77/13   77/17   77/21   77/25
TIMB  004521      77/14   77/18   77/26   77/32
TIMC  004507      77/15   77/34   77/42
TIN1, 004347      72/41   72/55
TIN2, 004350      72/30   72/56
TINC, 004262      72/21   72/40
TIND  004273      72/11   78/09
TINM, 004327      72/34   72/39
TINN, 004331      72/41
TINO  004272      65/40   72/10
TINR, 004265      72/04   72/44
TINS, 004302      72/18   72/29
TINW, 004305      72/21   72/38   72/54
TINX, 004264      72/03   72/20   72/32
TLIT  002431      43/26   45/26   45/46   48/29
TRET  001376      28/01   28/04   28/07   28/09   28/17   28/19   28/57
TTWAI 005444      92/19   92/29   92/48   93/05   93/18   93/25
TUMBL 004532      75/09   78/07
TUN   000567      18/37   19/02
TUNRE 000577      19/02   19/09   19/10
TYMCK 002531      43/33   44/04   48/06
TYME  004457      74/26   75/06   77/08
TYM,1 002545      48/10   48/18
TYM,2 002563      48/22   48/25   48/32
TYPAC 006317      16/10   26/48   27/03   29/34   60/31   60/35   60/47
                  87/06   87/08
TYPDE 006315      16/09   28/08
TYPE  004220      11/32   72/12   70/15   71/01   72/02   90/56
TYPE1 004231      71/10   71/23   71/27
TYPE2 004244      71/05   71/09   71/21
TYPRE 004261      71/01   71/20   71/34
TYPZ1 006320      16/11   18/13   19/20   23/24   27/16   28/03   47/24
                  47/28   65/07   65/31   85/22   85/26   85/30   85/34
                  85/38   87/04   87/13   87/35
UBP   000255      14/19   32/09   49/18   50/10
UBUFF 006615      14/19   106/04
UMSK  000301      14/36   18/30   32/21
UNIT  000174      12/26   19/03   24/08   25/16   25/37   26/05   27/24
                  28/02   29/08   29/10   29/17   29/18   32/27   33/33
                  38/39   40/32   46/25   46/23   48/13   48/27   49/28
                  53/11   53/27   54/13   54/26   55/09   55/23   56/24
                  56/32   57/10   57/40   57/47   58/09   58/37   59/02
                  62/04   62/11   60/40   61/23   61/29   62/14   62/32
                  85/18   85/37
UNTDN 000032      11/26   13/17   13/19   54/29   55/27
UNTIN 000026      11/22   22/05   26/14   54/14   55/10   57/11   62/15
UNTSK 000045      12/01   22/09
UPSEE 002423      43/16   43/31   44/09   44/10   45/09   45/40   47/23
VAR   007401      14/21   14/22   106/06
VARED 002261      14/22   51/14
VARET 005252      88/04   88/11   88/16
VARPT 000263      14/24   86/12   88/05   88/06   88/13
```