
RCSL No: 44-RT1844

Edition: January 1979

Author: H.O. Hansen

Title:

FDC 705 Testprogram

(Rev. 1.04)

Keywords:

RC3600 Floppy Disk Controller, Stand Alone Testprogram.

Abstract:

This program is a diagnostic test to check the controller FDC 705, and the main-functions of the disc drive.

(132 printed pages)

Copyright © 1982, A/S Regnecentralen af 1979
RC Computer A/S
Printed by A/S Regnecentralen af 1979, Copenhagen

Users of this manual are cautioned that the specifications contained herein are subject to change by RC at any time without prior notice. RC is not responsible for typographical or arithmetic errors which may appear in this manual and shall not be responsible for any damages caused by reliance on any of the materials presented.

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63

RCSL: 44-RT1844
AUTHOR: H.O.HANSEN
EDITED: 79.01.31

PROGRAM

FDC 705 TESTPROGRAM
(REV. 1.04)

KEYWORDS: RC3600 FLOPPY DISC CONTROLLER, STAND ALONE
TESTPROGRAM.

ABSTRACT: THE PROGRAM IS A DIAGNOSTICTEST TO CHECK THE
CONTROLLER FDC705, AND THE MAIN-FUNCTIONS OF
THE DISC-DRIVE.

ASCII TAPE: RCSL: 44-RT1845 (MAG.TAPE)
S-ABS.BINARY TAPE: RCSL: 44-RT1846 (WITH SELFSTART
AUTOLOAD HEADER)
BINARY CARDS: RCSL: 44-RT1847
PREWR. TEST DISCETTE: RCSL: 44-RT1759

```

01
02 ; *****
03 ; *
04 ; * REVISION HISTORY: *
05 ; *
06 ; * REV. DATE *
07 ; *
08 ; * 1.03 78.09.29 *
09 ; * 1.04 79.01.31 *
10 ; *
11 ; *****

```

```

12 ;
13 ;
14 ;:0. INDEX

```

15 ;	1. ABSTRACT	PAGE 2
16 ;	2. MACHINE REQUIREMENTS	PAGE 2
17 ;	3. SWITCH SETTINGS AND INIMESS	PAGE 3
18 ;	4. OPERATING PROCEDURE	PAGE 7
19 ;	5. PROGRAM DESCRIPTION	PAGE 8
20 ;	6. STRAPPINGS ETC.	PAGE 10
21 ;	7. STATUS BIT TABLE	PAGE 10
22 ;	8. BELONGING TESTEQUIPMENT	PAGE 10
23 ;	9. MESSAGES FROM THIS TEST	PAGE 11
24 ;		
25 ;		
26 ;		
27 ;		
28 ;		
29 ;:1. ABSTRACT		

```

30 ;
31 ; THE TESTPROGRAM WILL, AFTER START IN ONE OF THE DIFFE-
32 ; RENT STARTADDRESSES, CHECK THE FUNCTIONS OF THE DISC-
33 ; CONTROLLER AND DETECT HARDWARE-FAILURES. A RUN IS ALWAYS
34 ; INITIATED WITH A SIMPLE BUS-TEST, FOLLOWED BY TESTS OF
35 ; THE HARDWARE IN INCREASING COMPLEXITY.
36 ; IT'S POSSIBLE TO SEPERATE AND RUN DIFFERENT PARTS OF
37 ; THE TEST. BY SELECTING THE BELONGING STARTADDRESS, YOU
38 ; MAY RUN THE COMPLETE TEST, TEST OF WRITE LOGIC, TEST OF
39 ; READ LOGIC ETC.
40 ;

```

```

41 ; * NOTICE, THAT SOME OF THE TESTS IMPLIES, THAT THE DRIVE
42 ; * IS LOADED WITH THE PRE-WRITTEN TEST-DISCETTE (SEE SEC. 8)
43 ;
44 ;
45 ;

```

```

46 ;:2. MACHINE REQUIREMENTS

```

47 ;	RC 3600 FAMILY PROCESSOR	CPU
48 ;	MINIMUM 16K READ/WRITE MEMORY	MEM
49 ;	TELETYPE OR OPERATORS CONTROL PANEL	TTY/OCP *NOTE
50 ;	LINEPRINTER, OPTIONALLY	LPT *NOTE
51 ;		
52 ;		

```

53 ; *NOTE: IF THIS DEVICES IS SET ONLINE AFTER PROGRAM
54 ; START, THE PROGRAM MUST BE RESTARTED TO GET OUTPUT.

```


10004 .MAIN

```
01 ; AFTER START:
02 ;
03 ; SELECT OPERATION: 0?
04 ;
05 ; IMPLEMENTED OPERATIONS:
06 ;
07 ; 0 - READ RANDOM SECTORS WITHOUT
08 ; DATACHECK.
09 ; 1 - WRITE RANDOM SECTORS.
10 ; 2 - WRITE AND CHECKREAD RANDOM
11 ; SECTORS.
12 ;
13 ; 405 WILL EXECUTE A PART OF THE DIAGNOSTIC-LOOPS
14 ; AND END UP WITH STARTING THE MULTI-DRIVE
15 ; RELEABILITY TEST (WRITE-CHECKREAD).
16 ;
17 ;
18 ; 406 TEST OF DISCETTE FORMAT FUNCTION. A "ROUGH"
19 ; DISCETTE IS FORMATED TO IHM-FORMAT AND THEN
20 ; VERIFIED.
21 ;
22 ; AFTER START:
23 ;
24 ; NO. OF PASSES: 1?
25 ;
26 ; 407 CHANGE EQUIPMENT PARAMETERS IN PROGRAM.
27 ;
28 ; AFTER START:
29 ;
30 ; FIRST DRIVE TO TEST: 0?
31 ; LAST DRIVE TO TEST: 0?
32 ; NO. OF SURFACES: 1?
33 ; SECTOR LENGTH: 128?
34 ;
35 ; 410 SET PROGRAM IN IDU MODE. I.E. THE POWER-SUP.-
36 ; VOLTAGE WILL FROM NOW BE CONTROLLED BY THE
37 ; PROGRAM BY USE OF IDU 701.
38 ;
39 ; AFTER START:
40 ;
41 ; IDU DEV.CODE: 5?
42 ;
43 ; 411 CLEAR IDU MODE.
44 ;
45 ;
46 ; 1400 RESTART PROGRAM AS WHEN LOADED, BUT NOW AN-
47 ; NOUNCING MEM. SIZE, CPU-TYPE ETC
48 ; 2202 GET A NEW PRINTER ALPHABETH (SEE 4.1.1)
49 ; 2204 SET TO 64K WORDS MODE, MEM SIZE ?
50 ; 2206 SET TO 32K WORDS MODE, MEM SIZE ?
51 ; 2210 EXAMINE MEMORY
52 ; 2212 DEPOSIT MEMORY
53 ; 2214 TROUBLE BREAKPOINT HALT
54 ; 2216 TROUBLE BREAKPOINT LOOP REPORT
55 ; 2220 TROUBLE BREAKPOINT RESET
56 ; 2222 START BINARY LOADER, READ FROM PTR/TTI (SWO)
57 ;
58 ;
59 ; IDU-CONTROL:
60 ;
61 ; THE TESTS SA 400,401,405 MAY BE RUN IN IDU-MODE.
62 ; STARTING OTHER TESTS WILL AUTOMATICLY CLEAR
63 ; THE IDU-MODE.
```

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54

3.2 CONTROL BY SWITCH SETTING, THE STATE IS REPORTED
ON TELETYPE AND LINEPRINTER.

SW0: LOOP IN TEST IN CASE OF ERROR.
SW10: INHIBIT PRINTOUT IN TESTLOOP-PROGRAM.
SW11: PRINT FAILURE RATE IN TESTLOOP-PROGRAM.
SW12: NO HALT IN CASE OF ERROR IN TESTLOOP-PROGRAM.
SW13: WAIT AFTER DIS MESSAGE. DIS IS THE 16 CHAR.
SW14: HALT AFTER DIS MESSAGE. DISPLAY AT OPERATOR.
SW15: CLEAR DIS AFTER MESSAGE. CONTROL PANEL. OK.

* NOTICE: IF ANY OF SW 1 TO 9 ARE SET, THE CONTROL
* BY SW 13 TO 15 IS DISABLED.

3.3 START

OF PROGRAM AFTER LOADING:
SET SWITCHES TO CONTROL. (3.2).
ANSWER START ADDRESS QUESTION.

3.3A RESTART

THE BEST WAY TO INSURE CORRECT SWITCH-SETTINGS
AFTER A RESTART:

RESET
SET SWITCHES TO START ADDR. (3.1)
EXAMINE
SET SWITCHES TO CONTROL. (3.2)
CONTINUE.
THE START ADDR IS LISTED AT TTY, LPT AND DIS.

3.3B RESTART OF PROG. IF RC3603 CPU

SET DATA SWITCHES 0,10-15 UP (1)
SET "RESET PARITY ERROR" DOWN (ON)
PRESS AUTOLOAD AND RELEASE
SET "RESET PARITY ERROR" UP (OFF)
SET SWITCHES TO CONTROL (3.2)
ANSWER THE QUESTION WRITTEN ON TTY/OCP

3.3C POWER RESTART

OF PROGRAM (ONLY IF POWER MONITOR OPTION):

SET KEY IN LOCK BEFORE REMOVING POWER
AFTER POWERING UP, THE PROGRAM WRITES:

POWER
ACTUAL PROGRAM NAME
SET SWITCHES TO CONTROL. (3.2).
ANSWER START ADDRESS QUESTION

10006 .MAIN

```
01 ;
02 ;
03 ;
04 ;
05 ;
06 ;
07 ;
08 ;
09 ;
10 ;
11 ;
12 ;
13 ;
14 ;
15 ;
16 ;
17 ;
18 ;
19 ;
20 ;
21 ;
22 ;
23 ;
24 ;
25 ;
26 ;
27 ;
28 ;
29 ;
30 ;
31 ;
32 ;
33 ;
34 ;
35 ;
36 ;
37 ;
38 ;
39 ;
40 ;
41 ;
42 ;
43 ;
44 ;
```

3.4 LOADING MESSAGE:

AFTER LOADING THE PROGRAM WILL RUN THROUGH SOME
INITIALIZING CODE, ENDING UP WITH ANNOUNCING THE
PROGRAM-NAME. THE MEM.SIZE, CPU TYPE ETC., WHICH
ARE CALCULATED IN THIS CODE, MAY BE KNOWN BY
STARTING THE TESTPROGRAM IN LOC. 1400, AND THE
OUTPUT-FORMAT WILL BE LIKE THIS:

SWITCHES: 100073 ;THIS IS THE INITIAL STATE OF SW
CPU TYPE: 000016 ;THIS IDENTIFIES WHICH CPU/MEM
LAST LOC: 077777 ;THIS INFORMS THE USER: MEM-SIZE
BINARY LOADER OK ;THIS IS THE TS VERSION PLACED
;AT LAST LOC (MAX 32K) BUT
;SLIGHTLY MODIFIED AND INCL.
;BOOTSTRAP. THE LOADER HAS AN
;ERRORHALT AT XX7752 AND READY-
;HALT AT XX7676 IF NOT SELFSTRT.
ACTUAL PROG NAME ;THIS IS THE LOADED PROGRAM.
SET SWITCHES TO CONTROL.(3.2), STARTADDR 400 ?

3.5 START WITH BREAK OPTION (RC 3603).

RESET
SET SWITCHES TO HALT INSTR 063077
SET REGISTER SELECT TO 6
DEPOSIT INTO REGISTER
SET SWITCHES TO ADDRESS 000003
SET REGISTER SELECT TO 5
DEPOSIT INTO REGISTER
SET BREAK SWITCH TO ON
GO TO 3.3 RESTART WITH SA = 1400

3.6 CPUNO

FOR TIMING PURPOSE THE PROGRAM DETERMINES IN
WHICH CPU IT RESIDUES. IF IT FAILS THE PROGRAM
WILL TRY TO CONTINUE AFTER THE MESSAGE
"MISERABLE TIMING". IF IMPOSSIBLE THE PROGRAM
WILL ASK YOU TO IDENTIFY THE CPU WITH A NUMBER
BETWEEN 0 AND 6.

USE 2 FOR NOVA1200, RC3603-BREAK
4 FOR RC3603
5 FOR NOVA 2-16K
6 FOR NOVA 2-8K WHERE
THE MEMORY TYPE FOR THE FIRST 8K IS RELEVANT.

10007 .MAIN

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

24. OPERATING PROCEDURE

LOAD THE PROGRAM USING THE SELFSTARTING BINARY LOADER IN FRONT OF THE PAPER TAPE (ONLY IF RC3600, IN CASE OF A RC7000 YOU HAVE TO LOAD ANOTHER BIN. LOADER AND PASS-BY THE FIRST SEGM. ON TAPE. AFTER LOADING THE PROGRAM, SOME MESSAGES (SEC. 3.4) APPEARS, AND THE BINARY LOADER IS WILL BE CORRECTLY PLACED IN CORE. DURING THE INITIALIZING THE PROGRAM WILL FIND OUT IN WHICH CPU/MEM IT RESIDES (SEC 3.6).

NOW YOU HAVE TO SELECT THE WANTED TESTPROGRAM BY ANSWERING THE STARTADDRESS QUESTION (SEE SEC. 3.1). AFTER THAT SOME EQUIPMENT DEPENDING PARAMETERS HAVE TO BE SET, IF IT'S THE FIRST TESTSTART AFTER PROGRAM LOAD. THIS PARAMETERS WILL NOW STAY UNCHANGED UNTIL THE OPERATOR RESETS THEM BY STARTING IN SA 407.

IN CASE OF INPUTTING ANSWERS TO THE QUESTIONS BEYOND THE LIMITS, THE QUESTIONS ARE REPEATED. IF THE SUGGESTED ANSWER IS OK, ANSWER ONLY NL (RETURN). THE LAST DIGIT ANSWERED MAY BE REMOVED WITH KEY "RUROUT", ELSE, IF YOU TYPED WRONG, THE QUESTION IS REPEATED BY GIVING DIGITS UNTIL LIMIT IS PASSED.

THE PROGRAM CONTAINS AN ASCII PRINTER TABLE. IF THE PRINTER HAS ANOTHER DRUM ALPHABETH READ FOLLOWING:

24.1 PRINTER ALPHABETH CHANGE:

AT ANY TIME AFTER LOADING THIS PROGRAM IT IS POSSIBLE TO CHANGE THE ALPHABETH USED ON THE LINEPRINTER. THERE ARE 2 METHODS:

24.1.1

GET ONE OF THE BUILD-IN ALPHABETHS:

START PROGRAM IN 2202
REMEMBER SWITCHES TO CONTROL. (3.2).
ANSWER ALPHABETH 0 (SEE EXISTING BELOW).
THE PROGRAM WILL RESTART AFTER CHANGING THE ALPHABETH.
ANSWER NEXT START ADDRESS.

10008 .MAIN

```
01
02          ;4.1.2                      GET -ADD ON- A NEW TAPE .
03          ;
04          ;LOAD THE NEW ALPHABETH FROM TAPE INTO CORE UPON THE PROG. BY
05          ;USE OF THE BINARY LOADER. THEN RESTART THE PROGRAM. THE CURRENT
06          ;TAPES WITH PRINTER TABLE ALPHABETH ARE DESCRIBED BELOW:
07          ;
08          ;EXISTING: #1    44-RT 535      ASCII
09          ;              #2    44-RT 529      RC STANDARD TYPE 71/78 STARTING .
10          ;              #3    44-RT 532      RC STANDARD TYPE 71/78 STARTING 0
11          ;                                  (SKEWED 4 POSITIONS)
12          ;              #4    44-RT 1213     PL 1, TYPE 70
13          ;
14          ;HOW TO PRODUCE A NEW TABLE:
15          ;
16          ;THE TABLE HAS 200 OKTAL (128 DECIMAL) BOXES. INPUT KEY
17          ;IS THE ASCII VALUE OF THE CHARACTER TO PRINT ADDED TO 2000.
18          ;THE RESULT IS ADDRESS OF A BOX. EACH BOX OCCUPIES A CORE WORD.
19          ;IT IS BUILT UP OF TWO FIGURES. THE FIRST IS THE CLASS OF THE
20          ;CHARACTER TO BE PRINTED: 0 FOR PRINT, 6 FOR BLIND. THE SECOND
21          ;IS THE CHARACTER VALUE AT THE PRESENT PRINT DRUM. BELOW
22          ;VALUE 40 OCTAL FOLLOWING CHARACTERS MAY BE USED: 11 TAB,
23          ;12 LF, 14 FF AND 15 CR. ALL OTHERS BELOW 40 WILL GIVE SPACE.
24          ;THE FIRST BOX SHOULD CONTAIN THE VALUE FOR THE NULL CHAR
25          ;AND THE LAST THE VALUE FOR THE DEL CHAR, WHICH BOTH NORMALLY
26          ;ARE BLIND. IF YOU COUNT 0,1,2, ,7,10, , THE BOX 101 SHALL
27          ;CONTAIN THE PRINT DRUM VALUE FOR AN A. IF THE DRUM DO NOT
28          ;HAVE SMALL LETTERS, FILL IN THE VALUE FOR BIG ONES. NOW PUNCH
29          ;AN ASCII TAPE LIKE THIS:
30          ;
31          ; .LOC 2000
32          ; .RDX 8      ;WHICH RDX YOU WANT
33          ; .TXTE?
34          ; <6><0>      ; (2000) FIRST BOX, BOX 0
35          ; <6><0>
36          ; .
37          ; .
38          ; .
39          ; <0><101>    ; (2101) BOX 101 FOR A. FOR ASCII DRUM
40          ; .          ; 101 IS USED, FOR TYPE 71 137 IS USED.
41          ; .
42          ; .
43          ; <6><0>?    ; (2177) BOX 177, LAST
44          ; .RDX 8
45          ; .END 101
46          ;
47          ;PRODUCE A BINARY TAPE AND LOAD THIS TO MEMORY WITH
48          ;BINARY LOADER AFTER LOADING OF MAIN PROGRAM.
49          ;
50          ;
51          ;
52          ;
53          ;5. PROGRAM DESCRIPTION
54          ;
55          ;          THIS SECTION CONTAINS:          1. TESTLOOP FAILURE RATE
56          ;                                                  2. STRUCTURE OF PROGRAM
57          ;
58          ;
59          ;          5.1 TESTLOOP FAILURE RATE.
60          ;
61          ;THERE ARE TWO DIFFERENT WAYS TO USE THE ROUTINES FOR TESTLOOP:
62          ;SINGLE OR MULTIPLE ERRORHALT:
```

10009 .MAIN

```
01
02 ;SINGLE:          SETP1          MULTIPLE:  SETP1
03 ;              ERRORHALT        ERRORHALT
04 ;              LOOP            ERRORHALT
05 ;              ERRORHALT
06 ;              LOOP
07 ;
08 ;IN CASE OF A CONSTANT ERROR THE RATE WILL BE PRINTED
09 ;THIS WAY:
10 ;
11 ;PC XXXXXX 100 %                PC XXXXXX 300 %
12 ;WERE THE LAST IS A MULTIPLE OF 100 %.
13 ;
14 ;THERE ARE THREE POSSIBILITIES FOR THE NUMBER OF LOOPS IN A
15 ;CYCLE, I. E. HOW MANY TIMES THE PROGRAM RUNS THROUGH THE
16 ;INSTRUCTIONS BETWEEN SETP1 AND LOOP. IF SWITCH 0 IS 0 FIRST
17 ;TIME AN ERROR IS DETECTED THE PROGRAM PROCEEDS TO NEXT
18 ;INSTRUCTION AFTER LOOP.
19 ;
20 ;NUMBER OF LOOPS:          SETP0:  1
21 ;                          SETP1:  10
22 ;                          SETP2:  100
23 ;
24 ;IF THE ERROR IS NOT CONSTANTLY IT IS POSSIBLE TO SEE THESE
25 ;FAILURE RATES IF SINGLE OPERATION:
26 ;
27 ;SETP0: 100 % ERROR IN THE ONE LOOP.
28 ;SETP1: 100,50,33,25,20,16,14,12,11,10 % FOR ERROR IN
29 ;      1, 2, 3, 4, 5, 6, 7, 8, 9, 10 -TH LOOP.
30 ;SETP2: LIKE SETP1, ONLY ADD 9...1 % FOR ERROR IN
31 ;      11...100 -TH LOOP.
32 ;
33 ;IF MULTIPLE OPERATION THE RATES DEPENDS ON HOW MANY OF THE
34 ;ERRORS ARE FOUND IN THE FIRST LOOP WITH ERROR:
35 ;      100 % COULD BE  1 ERROR IN FIRST LOOP
36 ;                    OR    2 ERRORS IN SECOND LOOP
37 ;                    OR    3 ERRORS IN THIRD LOOP ETC.
38 ;
39 ;IF SWITCH 0 IS 1 THE PROGRAM WILL REMAIN IN THE LOOP WITH
40 ;ERROR. FIRST TIME AN ERROR IS SEEN THE PROGRAM WILL HALT
41 ;(IF NOT SWITCH 12). THEN THE PROGRAM CONTINUES UNTILL ALL
42 ;NUMBER OF LOOPS ARE PERFORMED. THEN A NEW CYCLE IS
43 ;ENTERED CALLED ERRORCYCLE. AFTER EACH ERRORCYCLE A NEW IS
44 ;STARTED UNTILL SWITCH 0 IS SET TO 0.
45 ;
46 ;IN ALL CYCLES (FIRST OR ERROR) THE PC (PROGRAM COUNTER OF ERROR)
47 ;IS WRITTEN FIRST TIME AN ERROR IS SEEN AND IN MULTIPLE OPERATION
48 ;MORE THAN ONE ERRORHALT COULD WRITE THE PC. BUT ONLY WITHIN
49 ;THAT FIRST LOOP WITH ERROR. THE FAILURE RATE IS PRINTED WHEN
50 ;THE CYCLE IS FINISHED. THE PROGRAM ONLY HALTS IN THE FIRST CYCLE
51 ;(DEPENDING ON SW 12). THE ERRORCYCLES HAVE ANOTHER AMOUNT OF
52 ;LOOPS THAN FIRST CYCLE:
53 ;
54 ;NUMBER OF LOOPS:          FIRST CYCLE      ERROR CYCLE
55 ;                          SETP0:  1          10
56 ;                          SETP1:  10         100
57 ;                          SETP2:  100        100
58 ;
59 ;IN SINGLE OPERATION FAILURE RATE IS TELLING HOW OFTEN THE ERROR
60 ;APPEARS. IN MULTIPLE OPERATION FAILURE RATE SHOULD BE USED
61 ;VERY CAREFULLY.
```

10010 MAIN

```
01 ;
02 ; 5.2 STRUCTURE OF PROGRAM.
03 ;
04 ; THE PROGRAM IS A NORMAL DIAGNOSTIC PROGRAM, I. E. IN
05 ; CASE OF ERROR THE PROGRAM WILL HALT WITH AC3 = PC. THE
06 ; OPERATOR SHOULD THEN LOOK UP IN THE LISTING TO KNOW WHAT
07 ; WAS CAUSING THE ERRORHALT. THE PROGRAM INCLUDES ONLY FEW
08 ; SUBROUTINES TO MAKE IT EASIER FOR THE OPERATOR.
09 ; THE PROGRAM ALSO CONTAINS NORMAL TESTLOOP FACILITIES, SEE
10 ; 5.1. IN CASE OF SEVERAL ERRORHALTS ATTENTION SHOULD ONLY
11 ; BE GIVEN TO THE FIRST (THE LOWEST PC) BECAUSE THE REST
12 ; COULD BE DERIVED FROM THE FIRST. TIMING IN A LOOP IS
13 ; AFFECTED BY PRINTING FIRST ERROR, TOO.
14 ;
15 ;
16 ;
17 ;
18 ;6. STRAPPINGS ETC.
19 ;
20 ; THIS TEST DOESN'T DEMAND ANY SPECIAL STRAPPING.
21 ;
22 ;
23 ;7. STATUS BIT TABLE.
24 ;
25 ;
26 ; BIT MNEMONIC
27 ;
28 ; 0 HARDWARE ERROR
29 ;
30 ; 1 OFF LINE
31 ; 2 REMAINING SECTORS
32 ; 3 SEEK ERROR
33 ;
34 ; 4 DELETED DATA
35 ; 5 WRITE PROTECT
36 ; 6 ILLEGAL COMMAND
37 ;
38 ; 7 -
39 ; 8 BLOCK LENGTH ERROR
40 ; 9 DATA LATE
41 ;
42 ; 10 CRC ERROR
43 ; 11 -
44 ; 12 POSITION ERROR
45 ;
46 ; 13 -
47 ; 14 -
48 ; 15 -
49 ;
50 ;
51 ;
52 ;
53 ;
54 ;8. BELONGING TESTEQUIPMENT.
55 ;
56 ; A SPECIAL PRE-WRITTEN DISCETTE (CALLED TEST DISCETTE
57 ; OR TM-DISCETTE) HAS TO BE USED IN THE TESTS WHICH
58 ; CHECKS THE READ DATA LOGIC (STARTADDRESS 401 (AND 402 IF
59 ; LOOP GOOD IS SELECTED)).
60 ; NO SPECIAL TESTPLUGS OR CABLES HAS TO BE USED DURING THE
61 ; TEST.
```

!0011 .MAIN

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59

9. MESSAGES FROM THIS TEST

ALL MESSAGES ARE WRITTEN ON TTY, LPT AND THE 16 CHAR DIS
ON OPERATORS CONTROL PANEL. IT IS POSSIBLE TO ANSWER
QUESTIONS AT TTY OR NUK, NUMERIC KEYBOARD ON OPERATORS
CONTROL PANEL.

ABOUT THE NUMBERS:

- 0-5 DIGITS IS A DECIMAL NUMBER
RANGE -32768 TO -1 AND 0 TO 32767
- 6 DIGITS IS AN OCTAL NUMBER
- 0-6 DIGITS IS AN OCTAL NUMBER WITH
LEADING ZEROES SUPPRESSED, DON'T USE.
- 8 DIGITS IS A BINARY NUMBER.

MESSAGES: LPT/TTY & DIS, IF SPEC DIS MESSAGE: ()

MISERABLE TIMING, RUN RC 3600 INSTRUCTION TIMER TEST
(MISERABLE TIMING)

TIMING WITH BIG TOLERANCES, CPU
TYPE IS GUESSED.

RTC IS UNSTABLE,

RTC NOT RUNNING WITH CONSTANT
SPEED.

SET CPUND > AC2

HELP THE PROGRAM TO IDENTIFY CPU
SEE 3.5

SWITCHES: 100073

NO MESS AT DIS ABOUT SW. POSITION

CPU TYPE: 000016

LAST LOC. 077777

BINARY LOADER OK

RC3600 FDC-TEST

SET SWITCHES TO CONTROL, (3.2), STARTADDR 400 ?

SWITCHES: 100030

NO MESS AT DIS ABOUT SW. POSITION

000401 STARTADDR

ILLEGAL INTERRUPT FROM DEV.: XX

;DURING THE TEST AN INTERRUPT
;WAS RECEIVED FROM AN UNEXPECTED
;DEVICE. THE DEVICE IS CLEARED,
;AND UP TO 5 ILLEGAL INTERRUPTS
;ARE NEGLECTED BEFORE THE PRO-
;GRAM GIVES UP AND GOES TO IDLE
;MODE.

CURRENT DRIVE: X

;WHEN AN ERROR IS DETECTED BY
;PROGRAM AND IT RELATES TO A SPE-
;SIFIC DRIVE, THE DRIVE NO. IS
;ANNOUNCED IN THIS WAY.

037500 037503 000000 ERROR MESSAGE

AC0 AC1 AC2

PC 007464 100 %

(PC 007464 100 %) EXAMINE FOR AC'S AFTER HALT.

SWITCHES: 100070

NO MESS AT DIS ABOUT SW. POSITION
PRINT INHIBIT SWITCH SET.

10012 .MAIN

```
01
02      000150 .LOC 150
03
04      ; CONSTANTS AND PROCEDURE-CALLS USED BY OR AS EXTENSIONS FOR THE
05      ; TS-STANDARD-ROUTINES.
06
07      000061      FDC= 61
08      000061      XFDC= FDC
09      000061      DEV= XFDC
10
11
12 00130 000061 DEVICE: DEV      ;CURRENT DEVICE-CODE IN TEST.
13
14 00131 177777 INIMK: 177777   ;INIMK IS <> 0 AFTER PROG.LOAD, BUT IS
15                               ;SET TO 0 AFTER FIRST CALL OF SW-READ-
16                               ;ROUT.
17
18 00132 000010 SWREG: 000010   ;INITIAL SWITCH-SETTING. USED INSTEAD
19                               ;OF THE ACTUAL SWITCHES UNTIL PROC.
20                               ;"CSAMS" IS CALLED FIRST TIME.
21
22
23
24      ;FDC 705 PROCEDURE CALL DEFINITIONS
25
26 00133 010002 CLL01: PASET
27      006133      SETPA= JSR      *CLL01
28 00134 011355 CLL02: GERAN
29      006134      GRAND= JSR     *CLL02
30 00135 011411 CLL03: RTRSE
31      006135      RANTS= JSR     *CLL03
32 00136 011303 CLL04: RCALI
33      006136      RECAL= JSR     *CLL04
34 00137 012312 CLL05: SCHAU
35      006137      EXECUTE= JSR   *CLL05
36 00140 010050 CLL06: ADMPA
37      006140      PAADM= JSP     *CLL06
38 00141 010657 CLL07: CHDEV
39      006141      SETDV= JSR    *CLL07
40 00142 012004 CLL08: PATFL
41      006142      PTFLL= JSR    *CLL08
42 00143 012147 CLL09: IUNIT
43      006143      IDUNIT= JSR   *CLL09
44 00144 012103 CLL10: D0SET
45      006144      SETD0= JSR    *CLL10
46 00145 012051 CLL11: RSTAT
47      006145      WGSTATUS= JSR *CLL11
48 00146 011405 CLL12: STINT
49      006146      SETIN= JSR    *CLL12
50 00147 012204 CLL13: ICHCOM
51      006147      INIACH= JSR   *CLL13
52 00150 012026 CLL14: TRCAL
53      006150      CALTR= JSR    *CLL14
54 00151 011775 CLL15: PATCM
55      006151      COMPT= JSR    *CLL15
56 00152 010167 CLL16: IASK
57      006152      INIASK= JSR   *CLL16
58 00153 010474 CLL17: IACT
59      006153      IACTION= JSR  *CLL17
60 00154 012272 CLL18: ISCH
61      006154      INISCH= JSR  *CLL18
```

!0013 MAIN

01

02	00155	011645	CLL19:	MERO		
03		006155		ERR00=	JSR	#CLL19
04	00156	011661	CLL20:	MER1		
05		006156		ERR01=	JSR	#CLL20
06	00157	011675	CLL21:	MER2		
07		006157		ERR02=	JSR	#CLL21
08	00160	011711	CLL22:	MER3		
09		006160		ERR03=	JSR	#CLL22
10	00161	012133	CLL23:	SBREG		
11		006161		SETBREG=	JSR	#CLL23
12	00162	011626	CLL24:	XEHALT		
13		006162		ERR04=	JSR	#CLL24
14	00163	007606	CLL25:	IDUCT		
15		006163		IDUADM=	JSR	#CLL25

!0014 .MAIN

```
01
02 ;SAVELOCATIONS FOR AC'S AFTER CALL OF PROCEDURES
03 00164 000000 SAV0R: 0
04 00165 000000 SAV00: 0
05 00166 000000 SAV01: 0
06 00167 000000 SAV02: 0
07
08 00170 000000 SAV1R: 0
09 00171 000000 SAV10: 0
10 00172 000000 SAV11: 0
11 00173 000000 SAV12: 0
12
13
14 00174 000000 RET00: 0 ;GENERAL RETURN SAVE
15 00175 000000 RET01: 0
16 00176 000000 IRTSV: 0
17 00177 000000 GRRET: 0 ;LOOP-GROUP RETURN SAVE
18
19
20 ;CONSTANTS
21
22 00200 100000 .1B0: 100000
23 00201 020000 .1B2: 020000
24 00202 010000 .1B3: 010000
25 00203 002000 .1B5: 002000
26 00204 001000 .1B6: 001000
27 00205 000200 .1B8: 000200
28 00206 000040 .1B10: 000040
29 00207 000010 .1B12: 000010
30 00210 000004 .1B13: 000004
31
32
33
34
35 00211 000010 F0010: 000010
36 00212 000014 F0014: 000014
37 00213 000077 F0077: 000077
38 00214 000377 F0377: 000377
39 00215 060000 F060K: 060000
40
41
42 00216 000002 F0002: 2
43 00217 000003 F0003: 3
44 00220 000004 F0004: 4
45 00221 000005 F0005: 5
46 00222 000006 F0006: 6
47 00223 000013 F0011: 11.
48 00224 000032 F0026: 26.
49 00225 000046 F0038: 38.
50 00226 000050 F0040: 40.
51 00227 000062 F0050: 50.
52 00230 000100 F0064: 64.
53 00231 000200 F0128: 128.
54 00232 000400 F0256: 256.
55 00233 000536 F0350: 350.
56 00234 000620 F0400: 400.
57 00235 001750 F01K0: 1000.
58 00236 003720 F02K0: 2000.
59 00237 005670 F03K0: 3000.
60 00240 012574 F05K5: 5500.
61
62
63 00241 177773 F0M05: -5
```


0015 .MAIN

```
01
02 00242 012503 BASAD: CPSTU ;CHANNEL PROG. BASE ADDR.
03
04 00243 130261 RDCOM: 130261 ;READ COMMAND
05 00244 036261 WRCOM: 036261 ;WRITE COMMAND
06 00245 135261 VF.COM: 135261 ;VERIFY COMMAND
07 00246 000261 RECOM: 000261 ;RECALIBRATE COMMAND
08 00247 010261 LBCOM: 010261 ;LOAD BAD CYLINDER COMMAND
09 00250 040261 LCCOM: 040261 ;LOAD CODE CONVERTER COMMAND
10 00251 060261 CCCOM: 060261 ;CLEAR CODE CONVERTER COMMAND
11 00252 025261 TFCOM: 025261 ;TRACK FORMAT COMMAND
12 00253 022261 RICOM: 022261 ;READ ID-FIELD COMMAND
13 00254 014261 RRCOM: 014261 ;READ BAD CYLINDER COMMAND
14 00255 050261 RCCOM: 050261 ;READ CODE CONVERTER COMMAND
15 00256 121661 RTCOM: 121661 ;READ TRACK COMMAND (NO ADDR. FIELD SYNC)
16
17 00257 000000 ARBUF: 0 ;POINTER TO CUR. READ-BUFFER (RBUF0,
18 ;RBUF1,RBUF2 OR RBUF3).
19 00260 000000 AWRUF: 0 ;POINTER TO CUR. WRITE-BUFFER (WBUF0,
20 ;WBUF1,WBUF2 OR WBUF3)
21
22 00261 017072 CURBF: RBUF0
23 00262 017072 RBFADR: RBUF0
24 00263 017076 WBFADR: WBUF0
25
26 00264 111350 STMSK: 111350
27 00265 063146 DUPAT: 063146
28
29
30 00266 000000 CURUN: 0 ;CURRENT DRIVE UNIT
31 00267 000000 MINDR: 0 ;FIRST DRIVE TO TEST
32 00270 000000 MAXDR: 0 ;MAX DRIVE TO TEST (0,1,2,3)
33 00271 000000 MAXSU: 0 ;MAX SURFACE NO. (0,1)
34
35 00272 000000 SEMA0: 0 ;SEMAPHORE 0
36 00273 000000 CLCK0: 0 ;CLOCK 0
37
38 00274 000000 SEMA1: 0
39 00275 000000 CLCK1: 0
40
41 00276 000000 SEMA2: 0
42 00277 000000 CLCK2: 0
43
44 00300 000000 SEMA3: 0
45 00301 000000 CLCK3: 0
```

```

10016 .MAIN
01
02 00302 016375 AMEXF: MEX00
03 00303 016515 MEX10
04 00304 016635 MEX20
05 00305 016755 MEX30
06 00306 016425 AMEXS: MEX01
07 00307 016545 MEX11
08 00310 016665 MEX21
09 00311 017005 MEX31
10 00312 016427 AMEXT: MEX02
11 00313 016547 MEX12
12 00314 016667 MEX22
13 00315 017007 MEX32
14
15 00316 016233 IMUX0: MUEXR ;POINTER TO PROCEDURE "MULTI DRIVE READ"
16 00317 016270 IMUX1: MUEXW ;POINTER TO PROCEDURE "MULTI DRIVE WRITE"
17 00320 016346 IMUX2: MUEXC ;POINTER TO PROCEDURE "MULTI DRIVE CHECK-
18 ;READ".
19 ;NEXT POINTER
20
21 00321 000002 MAXOP: .-IMUX0-1
22 00322 006316 MOPE0: JSR &IMUX0 ;MULTI OPERATION 0
23 00323 000401 JMP .+1-.,1
24 00324 000000 0
25 00325 006317 MOPE1: JSR &IMUX1 ;MULTI OPERATION 1
26 00326 000401 JMP .+1-.,1
27 00327 000000 0
28 00330 006317 MOPE2: JSR &IMUX1 ;MULTI OPERATION 2
29 00331 000402 JMP .+2-.,1
30 00332 006320 JSR &IMUX2
31 ;NEXT MULTI OPERATION
32
33
34
35
36 00333 000000 SECSIZ: 0 ;SECTOR SIZE (OCTAL 200,400,1000)
37 00334 000000 MXSEC: 0 ;MAX SECTOR NO. (OCTAL 32,17,10)
38 00335 000114 MXTRA: 76. ;MAX LEGAL TRACK NO.
39 00336 000000 PASCO: 0 ;PASS COUNTER
40 00337 0000124 ATBEG: XSTAA ;START OF DEVICE CHANGE
41 00340 017102 ATEND: TSEND ;END OF PROGRAM
42 00341 0000044 IEFLG: EBFLG ;<< 0 IF ERROR SINCE LAST SETPX
43 00342 000000 INTAD: 0
44 00343 000000 SLOOP: 0 ;<< 0 IF SINGLE LOOP MODE
45 00344 000000 LPAOR: 0
46 00345 000000 MULMK: 0 ;<< 0 IF MULTI PROCESS MODE
47 00346 000000 IDUMK: 0 ;<< 0 IF IDU MODE
48 00347 000000 SYSCLK: 0
49 00350 007667 PWCOM: PCOMA
50 00351 007666 ISETP: SETPOW
51 00352 000203 CCOM: NIOC 3
52 00353 060303 PCOM: NIOP 3
53 00354 060103 SCOM: NIOS 3
54 00355 000000 SIZSV: 0
55 00356 000000 SECSV: 0
56
57 00357 000000 WORK1: 0 ;COMMON WORK-CELL

```

!0017 .MAIN

01

02 000360 .LOC 360

03

04 00360 007315 FST00: FTES0

05 00361 007334 FST01: FTES1

06 00362 007361 FST02: FTES2

07 00363 007412 FST03: FTES3

08 00364 007421 FST04: FTES4

09 00365 007433 FST05: FTES5

10 00366 007462 FST06: FTES6

11 00367 010523 FST07: CDEVI

12 00370 007475 FST08: SIDUM

13 00371 007534 FST09: CIDUM

14 00372 007542 FST10: GENTD

15 00373 007310 FST11: FTES7

16

17 000376 .LOC 376

18

19 00376 006367 JSR #FST07 ;STARTADR. 376 - CHANGE FDC DEVICE CODE.

20 00377 006372 JSR #FST10 ;STARTADR. 377 - GENERATE TEST DISCETTE.

21 00400 006360 JSR #FST00 ;STARTADR. 400 - ALL FUNCTION TEST

22 00401 006361 JSR #FST01 ;STARTADR. 401 - TEST OF READ LOGIC

23 00402 006362 JSR #FST02 ;STARTADR. 402 - SINGLE TESTLOOP

24 00403 006363 JSR #FST03 ;STARTADR. 403 - TESTLOOP BUILDER

25 00404 006364 JSR #FST04 ;STARTADR. 404 - MULTI DRIVE RELIA. TEST

26 00405 006365 JSR #FST05 ;STARTADR. 405 - DIAGN. & RELIA. TEST

27 00406 006366 JSR #FST06 ;STARTADR. 406 - DISCETTE FORMAT TEST

28 00407 006373 JSR #FST11 ;STARTADR. 407 - CHANGE EQUIP. CONSTANTS

29 00410 006370 JSR #FST08 ;STARTADR. 410 - SET IDU MODE

30 00411 006371 JSR #FST09 ;STARTADR. 411 - CLEAR IDU MODE

!0018 .MAIN

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

***** FILENAME: FDCT2

007200 .LOC 7200

07200 141722 PROG: .TXTE !RC3600 FDC705-TEST

REV. 1.04<15><12>!

033083
030060
143240
141504
030267
026485
142724
152123
120240
120240
120240
151240
053305
120056
027251
132060
005215
000000

!0019 .MAIN

01

02	07223	007223	TROT8:	.
03	07224	012540		A000
04	07225	012545		A001
05	07226	012552		A002
06	07227	012561		A003
07	07230	012570		A004
08	07231	012577		A005
09	07232	012606		A006
10	07233	012614		A007
11	07234	012622		A008
12	07235	012631		A009
13	07236	012641		A010
14	07237	012656		B000
15	07240	012711		B001
16	07241	012733		B002
17	07242	012755		B003
18	07243	013022		C000
19	07244	013035		D000
20	07245	013142		D001
21	07246	013276		D002
22	07247	013430		E000
23	07250	013450		E001
24	07251	013506		E002
25	07252	013555		E009
26	07253	013627		E010
27	07254	013705		E003
28	07255	013777		E004
29	07256	014025		E005
30	07257	014132		E006
31	07260	014252		E007
32	07261	014301		F008
33	07262	014376		F000
34	07263	014457		F001
35	07264	014513		G000
36	07265	014625		H000
37	07266	014711		H001
38	07267	015035		H002
39	07270	015175		K000
40	07271	015540		K001
41	07272	015625		K002
42	07273	000000		0

TABLE WITH ADDRESSES OF ALL TESTLOOPS

10020 MAIN

```

01
02 07274 000000 TSBE6: 0 ;START OF PROGRAM
03
04 07275 007223 ATROT: TROT8
05 07276 010345 ANODB: NODRB
06 07277 012557 IGR00: GR000
07 07300 012655 IGR01: GR001
08 07301 013021 IGR02: GR002
09 07302 013052 IGR03: GR003
10 07303 013424 IGR04: GR004
11 07304 014366 IGR05: GR005
12 07305 014504 IGR06: GR006
13 07306 014622 IGR07: GR007
14 07307 015154 IGR10: GR010
15
16 07310 165000 FTES7: MOV 3,1 ;STARTADDRESS 407
17 07311 006072 CSAMS
18 07312 006764 JSR #ANODB ;SET EQUIP. PARAMS
19 07313 002401 JMP 4,+1
20 07314 000014A SWISA
21
22 07315 165000 FTES0: MOV 3,1 ;STARTADDRESS 400
23 07316 006072 CSAMS
24 07317 102440 SUB0 0,0 ;ACU:= 0
25 07320 006152 INIASK
26 07321 006153 SETPA
27 07322 006153 IACTION
28 07323 006754 FTS01: JSR #IGR00
29 07324 006754 JSR #IGR01
30 07325 006754 JSR #IGR02
31 07326 006754 JSR #IGR03
32 07327 006754 JSR #IGR04
33 07330 006754 JSR #IGR05
34 07331 006755 JSR #IGR07 ;EXECUTE WRITE TEST
35
36 07332 006140 PAADM
37 07333 000770 JMP FTS01
38
39 07334 165000 FTES1: MOV 3,1 ;STARTADDRESS 401
40 07335 006072 CSAMS
41
42 07336 102520 SUBZL 0,0
43 07337 006152 INIASK
44 07340 006153 SETPA
45 07341 006043 CCRLF
46 07342 006040 CMES8
47 07343 010125 LDPDS
48 07344 006104 CGTOK
49 07345 000402 JMP 4,+2
50 07346 000401 JMP 4,+1
51 07347 006153 IACTION
52 07350 006727 FTS10: JSR #IGR00
53 07351 006727 JSR #IGR01
54 07352 006727 JSR #IGR02
55 07353 006727 JSR #IGR03
56 07354 006727 JSR #IGR04
57 07355 006727 JSR #IGR05
58 07356 006727 JSR #IGR06
59
60 07357 006140 PAADM
61 07360 000770 JMP FTS10

```

!0021 .MAIN

```
01
02 07361 165000 FTES2: MOV      3,1      ;STARTADDRESS 402
03 07362 006072      CSAMS
04 07363 020216      LDA      0,F0002
05 07364 006152      INIASK
06 07365 102440      SUBO    0,0
07 07366 040346      STA     0,IDUMK ;CLEAR IDU MODE
08
09 07367 006043 FTES20: CCRLF                    ;ASK FOR LOOP-START ADDRESS
10 07370 006040      CMES
11 07371 010067      SLADR
12 07372 006104      CGTOK
13 07373 000774      JMP     FTS20
14 07374 000775      JMP     FTS20
15 07375 030022A     LDA     2,DIGIN
16 07376 036677     LDA     3,AAITROT
17 07377 175400 FTES21: INC     3,3
18 07400 025400     LDA     1,0,3
19 07401 125015     MOV    1,1,SNR
20 07402 000765     JMP    FTS20 ;ADDRESS NOT IN TABLE
21 07403 146414     SUBO   2,1,SZK ;SKIP IF ADDRESS OK
22 07404 000773     JMP    FTS21
23 07405 050344     STA     2,LPADR
24 07406 102000     ADC     0,0
25 07407 040343     STA     0,SLOOP ;SET SINGLE LOOP MARK
26 07410 006153     IACTION
27 07411 092344     JMP    &LPADR ;GO AND EXECUTE LOOP
28
29
30 07412 020217 FTES3:  LDA     0,F0003 ;STARTADDRESS 403
31 07413 006152     INIASK
32 07414 102440     SUBO    0,0
33 07415 040346     STA     0,IDUMK ;CLEAR IDU MODE
34 07416 006153     IACTION
35 07417 002401     JMP    &.+1
36 07420 012363     LHSTR
37
38
39 07421 165000 FTES4:  MOV     3,1      ;STARTADDRESS 404
40 07422 006072     CSAMS
41 07423 020220     LDA     0,F0004
42 07424 006152     INIASK
43 07425 102000     ADC     0,0
44 07426 040345     STA     0,MULMK ;SET MULTI PROCESS MARK
45 07427 102440     SUBO    0,0
46 07430 040346     STA     0,IDUMK ;CLEAR IDU-MODE
47 07431 002401     JMP    &.+1
48 07432 016147     Z000
49
50
51 07433 165000 FTES5:  MOV     3,1      ;STARTADDRESS 405
52 07434 006072     CSAMS
53 07435 020221     LDA     0,F0005
54 07436 006152     INIASK
55 07437 006153     IACTION
56
57 07440 006637 FTES50: JSR    #IGR00
58 07441 006637     JSR    #IGR01
59 07442 006637     JSR    #IGR02
60 07443 006637     JSR    #IGR03
61 07444 006637     JSR    #IGR04
62 07445 006637     JSR    #IGR05
```

10022 .MAIN

```
01
02 07446 020266 LDA 0,CURUN
03 07447 101400 INC 0,0 ;INCREMENT TO NEXT DRIVE
04 07450 040266 STA 0,CURUN
05 07451 024270 LDA 1,MAXDR
06 07452 106452 SUBZ0 0,1,SZC ;SKIP IF NEW UNIT > MAX DRIVE
07 07453 000765 JMP FTS50
08 07454 020267 LDA 0,MINDR
09 07455 040266 STA 0,CURUN
10 07456 102000 ADC 0,0
11 07457 040345 STA 0,MULMK
12 07460 002401 JMP 4,+1
13 07461 016147 ZUUU
14
15
16 07462 165000 FTES6: MOV 3,1 ;STARTADDRESS 406
17 07463 006072 CSAMS
18 07464 020222 LDA 0,FD006
19 07465 006152 INIASK
20 07466 102440 SURO 0,0
21 07467 040346 STA 0,IDUMK ;CLEAR IDU MODE
22 07470 006153 SETPA
23 07471 006153 IACTION
24
25 07472 006615 FTS60: JSR AIGR10
26 07473 006140 PAADM
27 07474 000776 JMP FTS60
28
29
30 07475 165000 SIDUM: MOV 3,1 ;STARTADDRESS 410
31 07476 006072 CSAMS
32 07477 102000 ADC 0,0
33 07500 040346 STA 0,IDUMK
34 07501 006071 CQUES ;ASK FOR IDU DEV.CODE
35 07502 007576 TDUDV
36 07503 007576 IDUDV
37 07504 007575 DEFID
38 07505 006053 CTDEC
39 07506 006057 CDDEC
40 07507 006104 CGTOK
41 07510 000462 JMP .+2
42 07511 000770 JMP .-8.
43 07512 0200224 LDA 0,DIGIN
44 07513 024213 LDA 1,F0077
45 07514 123400 AND 1,0
46 07515 124000 COM 1,1
47 07516 030352 LDA 2,CCOM
48 07517 133400 AND 1,2
49 07520 113000 ADD 0,2
50 07521 050352 STA 2,CCOM
51 07522 030353 LDA 2,PCOM
52 07523 133400 AND 1,2
53 07524 113000 ADD 0,2
54 07525 050353 STA 2,PCOM
55 07526 030354 LDA 2,SCOM
56 07527 133400 AND 1,2
57 07530 113000 ADD 0,2
58 07531 050354 STA 2,SCOM
59 07532 002401 JMP 4,+1
60 07533 0000148 SWISA
```


!0023 .MAIN

```
01
02 07534 165000 CI00M: MOV      3,1      ;STARTADDRESS 411
03 07535 006072      CSAMS
04 07536 102440      SUHO      0,0
05 07537 040346      STA      0, IDUMK
06 07540 002401      JMP      A,+1
07 07541 000014A     SWISA
08
09
10 07542 006071 GENTD: CGUES      ;STARTADDRESS 377
11 07543 010152      GENDR
12 07544 010152      GENDR
13 07545 007574      DEFGD
14 07546 006053      CTDEC
15 07547 006057      CDDEC
16 07550 006104      CGTOK
17 07551 000402      JMP      .+2
18 07552 000770      JMP      .-8.
19 07553 020022A     LDA      0, DIGIN
20 07554 042417      STA      0, RGENDA
21 07555 006415      JSR      RGENNS
22 07556 102440      SUHO      0,0
23 07557 040345      STA      0, SLOOP
24 07560 040345      STA      0, MULMK
25 07561 040346      STA      0, IDUMK
26 07562 040266      STA      0, CURUN
27 07563 006153      IACTION
28
29 07564 020211      LDA      0, F0010
30 07565 040152      STA      0, SWREG
31 07566 006405      JSR      RGENLO
32 07567 002401      JMP      A,+1
33 07570 000014A     SWISA
34
35 07571 015661 GENLO: GR011
36 07572 010415 GENNS: NOSRF
37 07573 015665 GENDA: LU00A
38 07574 000000 DEFGD: 0
39 07575 000005 DEFID: 5
40 07576 042311 IDUDV: .TXTE !IDU DEV.CODE: !
41      120125
42      142504
43      027126
44      147705
45      142504
46      120072
47      000000
```

!0024 MAIN

```
01
02 ;PROCEDURE CALL: IDUADM
03 ;
04 ;
05 ;ENTRY: AC1= CURRENT PASSCOUNTER
06 ;
07 ;RETURN: CALL+1 IF POWER-VOLTAGE IS CHANGED, AND A NEW
08 ; SERIES OF PASSES HAS TO BE RUN
09 ; CALL+2 IF ALL POWER-VOLTAGES IS FINISHED.
10 ; CALL+3 IF A PASS MORE HAS TO BE EXECUTED AT SAME
11 ; POWER-VOLTAGE.
12 ; CALL+4 IF IDU IS NOT USED.
13 ;
14 ;DESTROYED: AC3
15
16 07606 054170 IDUCT: STA 3,SAV1P
17 07607 050175 STA 2,SAV12
18 07610 044172 STA 1,SAV11
19 07611 040171 STA 0,SAV10
20 07612 054346 LDA 3,IDUMK
21 07613 175015 MOVW 3,3,SNR
22 07614 000441 JMP IDU04 ;RETURN CALL+4
23
24 07615 060277 INIDS
25 07616 054215 LDA 3,F060K
26 07617 054347 STA 3,SYSCLK ;RE-INITIALIZE SYS-CLOCK.
27 07620 054444 LDA 3,IDUPA ;NO OF PASSES BEFORE VOLTAGE SHIFT.
28 07621 156052 ADCW 1,3,SZC ;SKIP IF PASS-COUNTER >= AC3
29 07622 000454 JMP IDU03 ;RETURN CALL+3
30
31 07623 054556 LDA 3,APASB
32 07624 125440 SUBO 1,1
33 07625 045402 STA 1,2,3 ;CLEAR PASS COUNTER
34 07626 056350 LDA 3,APWCOM
35 07627 024352 LDA 1,CCOM
36 07630 156414 SUBO 1,3,SZR ;SKIP IF CURRENT VOLTAGE = NOM.
37 07631 000402 JMP IDUC1
38 07632 000425 JMP IDU02 ;RETURN CALL+2
39 07633 056350 IDUC1: LDA 3,APWCOM ;VOLTAGE HAS TO BE CHANGED
40 07634 024354 LDA 1,SCOM
41 07635 156414 SUBO 1,3,SZR ;SKIP IF CURRENT VOLTAGE = NOMINAL+5%
42 07636 000410 JMP +8.
43
44 07637 034353 LDA 3,PCOM
45 07640 056350 STA 3,APWCOM
46 07641 004425 JSR SETPOW ;SET VOLTAGE = NOMINAL - 5%
47
48 07642 006043 CCRLF
49 07643 006040 CMESS
50 07644 007752 POWM2
51 07645 000413 JMP IDU01 ;RETURN CALL+1
52
53 07646 054352 LDA 3,CCOM
54 07647 056350 STA 3,APWCOM
55 07650 004416 JSR SETPOW ;SET VOLTAGE = NOMINAL
56
57 07651 006043 CCRLF
58 07652 006040 CMESS
59 07653 007703 POWM0
60 07654 000404 JMP IDU01 ;RETURN CALL+1
```

10025 .MAIN

01

02 07655 010170 IDU04: ISZ SAV1R
03 07656 010170 IDU03: ISZ SAV1R
04 07657 010170 IDU02: ISZ SAV1R
05 07660 030175 IDU01: LDA 2,SAV12
06 07661 024172 LDA 1,SAV11
07 07662 020171 LDA 0,SAV10
08 07663 002170 JMP *SAV1R ;RETURN

09

10 07664 000003 IDUPA: 3 ;NO. OF PASSES BEFORE VOLTAGE SHIFT
11 07665 000000 SPOWR: 0

12

13 07666 054777 SETPOW: STA 3,SPOWR

14 07667 000000 PCOMA: 0 ;CHANGE VOLTAGE INSTRUCTION TO IDU IS
15 ;PLACED HERE.

16 07670 006061 CWAIT

17 07671 000234 FD400

18 07672 062677 IORST

19 07673 102440 SUBO 0,0

20 07674 006146 SETIN

21 07675 006061 CWAIT

22 07676 000236 FD2K0

23 07677 002766 JMP *SPOWR

24

25

26 07700 060277 YLORE: INTDS ;LOOP-REPORT EXTENSION. DISABLE

27 07701 002401 JMP *+1 ;INTERRUPT BEFORE MESSAGE.

28 07702 000002A XLORE

10026 .MAIN

```
01
02 07705 120252 POWM0: .TXTE  !* * * POWER VOLTAGE = NOMINAL * * *!
03      120252
04      120252
05      147520
06      142727
07      120322
08      147526
09      152314
10      043501
11      120305
12      120275
13      147516
14      144515
15      040516
16      120314
17      120252
18      120252
19      000252
20
21 07725 120252 POWM1: .TXTE  !* * * POWER VOLTAGE = NOMINAL + 5% * * *!
22      120252
23      120252
24      147520
25      142727
26      120322
27      147526
28      152314
29      043501
30      120305
31      120275
32      147516
33      144515
34      040516
35      120314
36      120055
37      122465
38      125240
39      125240
40      125240
41      000000
42
43 07752 120252 POWM2: .TXTE  !* * * POWER VOLTAGE = NOMINAL - 5% * * *!
44      120252
45      120252
46      147520
47      142727
48      120322
49      147526
50      152314
51      043501
52      120305
53      120275
54      147516
55      144515
56      040516
57      120314
58      120055
59      122465
60      125240
61      125240
62      125240
63      000000
```

0027 .MAIN

01

02 07777 000000 NOPAS: 0

03 10000 000011AXPAS: XPASS

04 10001 000010APASB: PASSB

05

06

07

08

09

;PROCEDURE CALL: SETPA

10

;PROCEDURE TO INITIALIZE THE PASS-ADMINISTRATOR.

11

12

13 10002 054174 PASET: STA 3,RETUO

14 10003 030776 LDA 2,APASB ;CLEAR STANDARD ROUT. PASS COUNTERS

15 10004 102520 SUBZL 0,0

16 10005 041000 STA 0,0,2

17 10006 041001 STA 0,1,2 ;SET NO. OF PASSES BETWEEN MESS = 1

18 10007 102440 SUBO 0,0

19 10010 041002 STA 0,2,2 ;CLEAR PASS COUNTER

20 10011 020267 LDA 0,MINDR

21 10012 040260 STA 0,CURUN

22

23 10013 006071 RNMES: CQUES ;ASK FOR WANTED NO. OF PASSES

24 10014 010110 TNRUN

25 10015 010120 DNRUN

26 10016 010130 DFRUN

27 10017 006055 CTDEC

28 10020 006057 CDDEC

29 10021 006105 CGTDC

30 10022 000402 JMP .+2

31 10023 000770 JMP RNMES

32 10024 020022A LDA 0,DIGIN

33 10025 040752 STA 0,MOPAS ;STORE WANTED NO. OF PASSES

34 10026 040630 STA 0,IDUPA

35 10027 002174 JMP RETUO

10028 .MAIN

```
01
02      ;PROCEDURE CALL: PAADM
03      ;
04      ;PROCEDURE TO ADMINISTER THE EXECUTION OF THE WANTED NUMBER OF
05      ;PASS'S. RETURN IS DONE TO CALL+1 IF THE WANTED NO. OF PASSES
06      ;ISN'T EXECUTED YET, ELSE TO STARTADDRESS QUESTION-ROUTINE.
07
08 10030 054174 ADMPA: STA      3,RETUO
09 10031 060277          INTDS
10 10032 062677          IORST
11 10033 006061          CWAIT          ;IORST WAIT
12 10034 000237          FD3K0
13 10035 062677          IORST
14 10036 006061          CWAIT
15 10037 000235          FD1K0
16 10040 010266          ISZ      CURUN
17 10041 020266          LDA      0,CURUN
18 10042 024270          LDA      1,MAXDR
19 10043 106432          SUBZ0    0,1,SZC ;SKIP IF NEW UNIT > MAX DRIVE
20 10044 002174          JMP      ARETUO
21 10045 020267          LDA      0,MINDR
22 10046 040266          STA      0,CURUN
23 10047 006731          JSR      RAXPAS ;GO TO STANDARD PASS ADM. ROUT.
24 10050 030731          LDA      2,APASB
25 10051 025002          LDA      1,2,2 ;GET PASS COUNTER
26
27 10052 006163          IDUADM
28 10053 002174          JMP      ARETUO ;POWER VOLTAGE IS CHANGED
29 10054 000405          JMP      PASFI ;ALL COMPLETED.
30 10055 002174          JMP      ARETUO ;NEW PASS WITHOUT POWER CHANGE
31                          ;IDU NOT USED.
32 10056 020721          LDA      0,NOPAS
33 10057 122032          ADCZ0    1,0,SZC
34 10060 002174          JMP      ARETUO ;EXECUTE NEW PASS
35
36 10061 020211 PASFI: LDA      0,F0010
37 10062 040132          STA      0,SWREG ;DISABLE PANEL-SWITCH CONTROL
38
39 10063 006050          CHAAT          ;RUN COMPLETED
40 10064 006050          CHAAT
41 10065 002401          JMP      R.+1
42 10066 000014a        SWISA
43
44
45 10067 142523 SLADR: .TXTE  !SELECT LOOP STARTADDR. (OCTAL): !
46      142714
47      152303
48      146240
49      147717
50      120120
51      152123
52      151101
53      040724
54      042104
55      027322
56      024240
57      141717
58      040724
59      124714
60      120072
61      000000
```

10029 .MAIN

01

02 10110 147516 TNRUN: .TXTE !NO. OF PASSES: !

03 120056

04 145317

05 050240

06 051501

07 142525

08 055125

09 000240

10

11 10120 120243 ONRUN: .TXTE !2 OF PASS!

12 145317

13 050240

14 051501

15 000125

16

17 10125 147714 LDPDS: .TXTE !LOAD PRE-WRITTEN TESTDISCETTE(S),TYPE NL !

18 042101

19 050240

20 142722

21 153455

22 144722

23 152524

24 047505

25 152240

26 051705

27 042524

28 051711

29 142705

30 152524

31 024505

32 124525

33 152254

34 050151

35 120505

36 146116

37 000240

38

39 10152 144504 GENOR: .TXTE !DISCETTE FOR DRIVE NO: !

40 141525

41 152505

42 142724

43 145240

44 151517

45 042240

46 144722

47 142526

48 047240

49 055317

50 000240

51

52 10166 000001 OFRUN: 1

10050 MAIN

```
01
02          ;PROCEDURE CALL: INIASK
03
04 10167 054541 IASK: STA 3,IASKR
05 10170 040557 STA 0,STAID
06 10171 006073 CRESW          ;READ SWITCHES TO GET CUR. SETTING.
07 10172 006043 CCRLF
08 10173 020554 LDA 0,STAID
09 10174 054267 LDA 3,MINDR
10 10175 054266 STA 3,CURUN
11 10176 101005 MOV 0,0,SNR
12 10177 000421 JMP IAS00      ;CALLED FROM 400
13 10200 176520 SUBZL 3,3
14 10201 116415 SUB0 0,3,SNR
15 10202 000420 JMP IAS01      ;CALLED FROM 401
16 10203 175400 INC 3,3
17 10204 116415 SUB0 0,3,SNR
18 10205 000417 JMP IAS02      ;CALLED FROM 402
19 10206 175400 INC 3,3
20 10207 116415 SUB0 0,3,SNR
21 10210 000434 JMP IAS03      ;CALLED FROM 403
22 10211 175400 INC 3,3
23 10212 116415 SUB0 0,3,SNR
24 10213 000432 JMP IAS04      ;CALLED FROM 404
25 10214 175400 INC 3,3
26 10215 116415 SUB0 0,3,SNR
27 10216 000470 JMP IAS05      ;CALLED FROM 405
28
29 10217 000472 JMP IAS06      ;CALLED FROM 406 OR GREATER
30
31 10220 004521 IAS00: JSR NODRA
32 10221 004471 JSR IASKO
33
34 10222 004517 IAS01: JSR NODRA
35 10223 000467 JMP IASKO
36
37 10224 006071 IAS02: CQUES          ;ASK FOR WHICH DRIVE TO TEST
38 10225 010612 SELDR
39 10226 010612 SELDR
40 10227 010553 DEFSL
41 10230 006053 CTDEC
42 10231 006057 CDDEC
43 10232 006105 CGTDC
44 10233 000402 JMP .+2
45 10234 000770 JMP .-8.
46 10235 020022 LDA 0,DIGIN
47 10236 024217 LDA 1,F0003
48 10237 106433 SUBZ0 0,1,SNC
49 10240 000764 JMP .-12.
50 10241 040266 STA 0,CURUN
51 10242 004551 JSR NOSRF
52 10243 000447 JMP IASKO
53
54 10244 000446 IAS03: JMP IASKO
```


10031 MAIN

```

01
02
03 10245 004474 JSR NODRA
04 10246 006071 CQUES ;SELECT OPERATION
05 10247 010622 SELOP
06 10250 010622 SELOP
07 10251 010354 DEFOP
08 10252 006053 CTDEC
09 10253 006057 CDDEC
10 10254 006105 CGTDC
11 10255 000402 JMP .+2
12 10256 000770 JMP .-8.
13 10257 020022A LDA 0,DIGIN
14 10260 024521 LDA 1,MAXOP
15 10261 106453 SUBZ0 0,1,SNC ;SKIP IF INPUT <= MAXOP
16 10262 000764 JMP .-12.
17
18 10263 030452 IAS4A: LDA 2,OPPOI
19 10264 113000 ADD 0,2
20 10265 031000 LDA 2,0,2
21 10266 025000 LDA 1,0,2 ;AC1= WANTED OPERATION
22 10267 046302 STA 1,AMEXF
23 10270 046303 STA 1,AMEXF+1
24 10271 046304 STA 1,AMEXF+2
25 10272 046305 STA 1,AMEXF+3
26 10273 025001 LDA 1,1,2
27 10274 046306 STA 1,AMEXS
28 10275 046307 STA 1,AMEXS+1
29 10276 046310 STA 1,AMEXS+2
30 10277 046311 STA 1,AMEXS+3
31 10300 025002 LDA 1,2,2
32 10301 046312 STA 1,AMEXT
33 10302 046313 STA 1,AMEXT+1
34 10303 046314 STA 1,AMEXT+2
35 10304 046315 STA 1,AMEXT+3
36 10305 000405 JMP IASKO
37
38 10306 004453 IAS05: JSR NODRA
39 10307 020216 LDA 0,FDO02 ;SELECT OPERATION 2
40 10310 000753 JMP IAS4A
41
42 10311 004430 IAS06: JSR NODRA
43
44 10312 102440 IASKO: SUBO 0,0
45 10313 040343 STA 0,SLOOP ;CLEAR SINGLE LOOP MARK
46 10314 040345 STA 0,MULMK ;CLEAR MULTI PROCESS MARK
47 10315 002413 JMP RIASKR
48
49 10316 015167 FDATL: K000J
50 10317 015170 FSECT: K000K
51 10320 000032 FSEC1: 26.
52 10321 000200 FDTL1: 128.
53 10322 000017 FSEC2: 15.
54 10323 000400 FDTL2: 256.
55 10324 000010 FSEC3: 8.
56 10325 001000 FDTL3: 512.
57 10326 000200 FDLDE: 128.
58 10327 000000 STALD: 0
59 10330 000000 IASKK: 0
60 10331 000000 DRIDF: 0
61 10332 000001 SURDF: 1
62 10333 000000 DEFSL: 0
63 10334 000000 DEFOP: 0

```

0032 .MAIN

```

01
02 10335 010336 OPP01:  .+1          ;POINTER TO MULTI DRIVE OPERATION-SETS
03 10336 000322          MOPE0
04 10337 000325          MOPE1
05 10340 000330          MOPE2
06
07
08 10341 054357 NODRA:  STA      3,WORK1 ;ASK FOR FIRST DRIVE
09 10342 020333          LDA      0,SECSIZ
10 10343 101014          MOV0    0,0,SZR ;SKIP IF NOT DEFINED YET
11 10344 002357          JMP      4,WORK1
12
13 10345 054357 NODRB:  STA      3,WORK1
14 10346 006071          CQUES
15 10347 010553          FIDRV
16 10350 010553          FIDRV
17 10351 010331          DRIDF
18 10352 006053          CTDEC
19 10353 006057          CDDEC
20 10354 006105          CGTDC
21 10355 000402          JMP      .+2
22 10356 000770          JMP      .-8.
23 10357 020022A        LDA      0,DIGIN
24 10360 024220          LDA      1,F0004
25 10361 106433          SUBZ0   0,1,SNC ;SKIP IF ACC <= 4
26 10362 000760          JMP      NODRA+1
27 10363 040267          STA      0,MINDR
28 10364 040266          STA      0,CURUN
29 10365 054357          LDA      3,WORK1
30
31 10366 054357 LADRA:  STA      3,WORK1 ;ASK FOR LAST DRIVE
32 10367 006071          CQUES
33 10370 010566          LADRV
34 10371 010566          LADRV
35 10372 010331          DRIDF
36 10373 006053          CTDEC
37 10374 006057          CDDEC
38 10375 006105          CGTDC
39 10376 000402          JMP      .+2
40 10377 000770          JMP      .-8.
41 10400 020022A        LDA      0,DIGIN
42 10401 024217          LDA      1,F0003
43 10402 106433          SUBZ0   0,1,SNC ;SKIP IF ACC <= 3
44 10403 000764          JMP      LADRA+1
45 10404 024267          LDA      1,MINDR
46 10405 106032          ADCZ0   0,1,SZC ;SKIP IF LAST DR. >= FIRST DR.
47 10406 000761          JMP      LADRA+1
48 10407 040270          STA      0,MAXDR
49 10410 102440          SUB0
50 10411 040333          STA      0,SECSIZ
51 10412 034357          LDA      3,WORK1
52
53 10413 054357 NOSRF:  STA      3,WORK1
54 10414 020333          LDA      0,SECSIZ
55 10415 101014          MOV0    0,0,SZR
56 10416 002357          JMP      4,WORK1
57 10417 006071          CQUES          ;ASK FOR NO OF SURFACES
58 10420 010601          NOSUR
59 10421 010601          NOSUR
60 10422 010332          SURDF
61 10423 006053          CTDEC
62 10424 006057          CDDEC

```

10033 .MAIN

```
01
02 10425 006105 CGTDC
03 10426 000402 JMP .+2
04 10427 000704 JMP NOSRF
05 10430 020022A LDA 0,0IGIN
06 10431 101015 MOV@ 0,0,SNR
07 10432 000701 JMP NOSRF
08 10433 024210 LDA 1,F0002
09 10434 100433 SUBZ@ 0,1,SNC ;SKIP IF ACU <= 2
10 10435 000750 JMP NOSRF
11 10436 126520 SUBZL 1,1
12 10437 122400 SUB 1,0
13 10440 040271 STA 0,MAXSU
14
15 10441 006071 DATLA: CRUES ;ASK FOR SECTOR LENGTH
16 10442 010634 FDTFL
17 10443 010634 FDTFL
18 10444 010320 FOLDE
19 10445 006053 CTDEC
20 10446 006057 C0DEC
21 10447 006105 CGTDC
22 10450 000402 JMP .+2
23 10451 000770 JMP .-8.
24 10452 020022A LDA 0,0IGIN
25 10453 024645 LDA 1,FSEC1
26 10454 034645 LDA 3,FDTL1
27 10455 116415 SUB@ 0,3,SNR
28 10456 000411 JMP DATLO ;SEC. LENGTH = 128
29 10457 024643 LDA 1,FSEC2
30 10460 034643 LDA 3,FDTL2
31 10461 116415 SUB@ 0,3,SNR
32 10462 000405 JMP DATLO ;SEC LENGTH = 256
33 10463 024641 LDA 1,FSEC3
34 10464 034641 LDA 3,FDTL3
35 10465 116414 SUB@ 0,3,SNR
36 10466 000753 JMP DATLA ;ILLEGAL ANSWER
37 ;SEC. LENGTH = 512
38 10467 042627 DATLO: STA 0,AFDACL
39 10470 046027 STA 1,AFSECT
40 10471 040333 STA 0,SECS12
41 10472 044334 STA 1,MXSEC
42 10473 002357 JMP #WORK1
43
44 ;PROCEDURE CALL: IACTION
45
46 10474 054634 IACT: STA 3,IASKR
47 10475 062677 IORST
48 10476 006061 WAIT ;IORST WAIT
49 10477 000237 FD3KU
50 10500 062677 IORST
51 10501 006061 WAIT
52 10502 000235 FD1KU
53 10503 034417 LDA 3,LOPAD
54 10504 056415 STA 3,AXILOR ;SET LOOP REPORT ADDR=YLORE
55 10505 034334 LDA 3,SCOM
56 10506 056350 STA 3,APWCOM ;SET POWER VOLTAGE = NOMINAL+5%
57 10507 034346 LDA 3,IDUMK
58 10510 175015 MOV@ 3,3,SNR
59 10511 000405 JMP .+5
60 10512 006351 JSR #ISETP ;CHANGE POWER
```

10034 .MAIN

```
01
02 10513 006043      CCRLF
03 10514 006040      CMESS
04 10515 007725      POWM1
05 10516 102440      SUBO      0,0
06 10517 006146      SETIN      ;SET UP INTERRUPT ROUTINES
07 10520 002610      JMP      &IASKR
08
09 10521 000001HXILOR: ILORE
10 10522 007700 LOPAD: YLORE
11
12 10523 020211 CDEVI: LDA      0,F0010 ;ROUTINE TO CHANGE DEV.CODE OF FDC.
13 10524 040132      STA      0,SWREG
14
15 10525 006043      CCRLF
16 10526 006046      CDICL
17 10527 006040      CMESS
18 10530 010545      DEVCO
19 10531 006044      CDISP
20 10532 010545      DEVCO
21 10533 006050      CHAAT
22 10534 006104      CGTOK
23 10535 000766      JMP      CDEVI
24 10536 000765      JMP      CDEVI
25 10537 020022A     LDA      0,DIGIN
26 10540 024337      LDA      1,ATBEG
27 10541 030349      LDA      2,ATEND
28
29 10542 006141      SETDV
30
31 10543 002401      JMP      &.+1
32 10544 000014A     SWISA
33
34 10545 142504 DEVCO: .TXTE      !DEV.CODE= !
35      027126
36      147703
37      142504
38      120275
39      000000
40
41 10553 144706 FIDRV: .TXTE      !FIRST DRIVE TO TEST: !
42      051722
43      120324
44      151104
45      053311
46      120305
47      147724
48      152240
49      051705
50      035324
51      000240
52
53 10566 040714 LAORV: .TXTE      !LAST DRIVE TO TEST: !
54      152123
55      042240
56      144722
57      142526
58      152240
59      120317
60      142724
61      152123
62      120072
63      000000
```

0035 .MAIN

01
02 10601 147516 NUSUR: .TXTE !NO. OF SURFACES: !
03 120056
04 143317
05 051040
06 151125
07 040706
08 142703
09 035123
10 000240
11
12 10612 151104 SELDR: .TXTE !DRIVE TO TEST: !
13 053311
14 120305
15 147724
16 152240
17 051705
18 035324
19 000240
20
21 10622 142523 SELOP: .TXTE !SELECT OPERATION: !
22 142714
23 152303
24 147640
25 142520
26 040722
27 144724
28 047317
29 120072
30 000000
31
32 10634 142523 FDTFL: .TXTE !SECTOR LENGTH: !
33 152303
34 151317
35 146240
36 047305
37 152107
38 035110
39 000240
40
41 10644 125252 DWPRO: .TXTE !*** WRITE-PRO. IS ON!
42 120252
43 151327
44 152311
45 026705
46 151120
47 027317
48 144640
49 120123
50 047317
51 000000

10036 MAIN

```
01
02 ;PROCEDURE CALL: SETDV
03 ;
04 ;THE PROCEDURE WILL CHANGE THE DEVICECODE IN ALL I/O-INSTRUC-
05 ;TIONS TO THE DEVICE SPECIFIED IN "DEVICE" (PAGE ZERO). THE
06 ;CHANGE WILL TAKE PLACE IN LOCATIONS FROM LOC. SPECIFIED IN AC1
07 ;TO LOC. SPECIFIED IN AC2. THE NEW DEVICE-CODE WILL BE SET =
08 ;B10 - B15 OF AC0.
09 ;
10 ;ENTRY: AC0= NEW DEVICE CODE,AC1= FIRST LOCATION,AC2= LAST
11 ; LOCATION (LOC. "DEVICE" = CURRENT DEVICE NO., IT
12 ; HAS TO BE SET EQUAL TO THE ASSEMBLED DEVICE NO.
13 ; IN THE ASCII-SOURCE PROGRAM.)
14 ;RETURN: .+1
15 ;
16 ;DESTROYED: AC3
17
18 10657 054164 CHDEV: STA 3,SAV0R
19 10660 044166 STA 1,SAV01
20 10661 024213 LDA 1,FOU77 ;AC1:= 77
21 10662 107400 AND 0,1
22 10663 034130 LDA 3,DEVICE;LAST USED CODE FOR DEVICE
23 10664 166415 SUB# 3,1,SNR ;SKIP IF LAST USED CODE <> NEW CODE
24 10665 000435 JMP CIMOU ;CHANGE IS NOT NECESSARY - EXIT
25
26 10666 050167 STA 2,SAV02 ;CHANGE ALL I/O-INSTR. TO "DEVICE"
27 10667 040165 STA 0,SAV00
28 10670 024166 LDA 1,SAV01
29 10671 044433 STA 1,INSPOI;FIRST ADDRESS TO EXAMINE
30 10672 132420 SUBZ 1,2
31 10673 151400 INC 2,2
32 10674 050431 STA 2,INSCOU;# OF INSTRUCTIONS TO EXAMINE
33 10675 030213 LDA 2,FOU77 ;AC2:= 77
34 10676 143400 AND 2,0
35 10677 040130 STA 0,DEVICE;UPDATE DEVICE
36 10700 024215 LDA 1,FO60K ;AC1:= 060000
37 10701 123000 ADD 1,0 ;AC0:= NEW CODE WITH BIT 1,2
38 10702 167000 ADD 3,1 ;AC1:= LAST USED CODE WITH BIT 1,2
39
40 10703 010421 NXINS: ISZ INSP01 ;INCREMENT INSTRUCTION POINTER
41 10704 032420 LDA 2,INSPOI;GET INSTRUCTION
42 10705 034421 LDA 3,IOMSK ;AC3:= 160077
43 10706 014417 JSZ INSCOU
44 10707 157401 AND 2,3,SKP ;AC3:= CURR. INSTR. WITH OUTMASKED BIT
45 ;3,4,5,6,7,8,9
46 10710 000410 JMP SDOUT ;ALL INSTRUCTIONS EXAMINED - EXIT
47 10711 166414 SUB# 3,1,SZR ;SKIP IF INSTR. IS I/O-INSTR TO LAST
48 ;USED DEVICE
49 10712 000771 JMP NXINS ;NOT A I/O-INSTR. GET NEXT
50 10713 034414 LDA 3,FUMSK ;AC3:= 017700
51 10714 173400 AND 3,2 ;AC2:= CURRENT FUNCTIONBITS
52 10715 113000 ADD 0,2 ;AC2:= NEW I/O-INSTR
53 10716 052406 STA 2,INSPOI;STORE INSTR.
54 10717 000764 JMP NXINS ;GET NEXT INSTR.
55 10720 020165 SDOUT: LDA 0,SAV00
56 10721 030167 LDA 2,SAV02
57 10722 024166 CIMOU: LDA 1,SAV01
58 10723 002164 JMP ASAV0R
59
60 10724 000000 INSP01: 0 ;POINTER TO CURRENT INSTR.
61 10725 000000 INSCOU: 0 ;INSTRUCTION COUNTER
62 10726 160077 IOMSK: 160077
63 10727 017700 FUMSK: 017700
```

```

0057 .MAIN
01 ; COMMON INTERRUPT SERVICE ROUTINE.
02
03 10730 054520 INTSV: STA 3,ISAV3
04 10731 050520 STA 2,ISAV2
05 10732 044520 STA 1,ISAV1
06 10733 040520 STA 0,ISAV0
07 10734 101200 MOVR 0,0
08 10735 040517 STA 0,ISAVC
09 10736 034000 LDA 3,0
10 10737 054176 STA 3,IRTSV ;SAVE RETURN ADDRESS
11 10740 034021A LDA 3,POWZE
12 10741 054000 STA 3,0 ;RESTORE POWER-UP INSTR.
13 10742 063777 SKPDZ CPU ;SKIP IF NOT POWER INTERRUPT
14 10743 000474 JMP PwINT
15 10744 063610 SKPDN TTI
16 10745 000411 JMP GTIDV ;NOT TTI
17 ;TTI INTERRUPT - RESTART PROGRAM
18 10746 062677 RSPRO: IORST
19 10747 034352 LDA 3,CCOM
20 10750 056350 STA 3,RPWCOM
21 10751 034346 LDA 3,IDUMK
22 10752 175014 MOVW 3,3,SZR
23 10753 006351 JSR R1SETP
24 10754 002401 JMP R.+1 ;RESTART PROG.
25 10755 000014A SWISA
26
27 10756 071477 GTIDV: INTA 2
28 10757 034213 LDA 3,FOU77
29 10760 173400 AND 3,2
30 10761 034212 LDA 3,FOU14
31 10762 156413 SUBW 2,3,SNR
32 10763 000516 JMP RTCIN ;CLOCK INTERRUPT
33 10764 050471 STA 2,ILDEV
34 10765 034130 LDA 3,DEVICE
35 10766 156413 SUBW 2,3,SNR
36 10767 000436 JMP INTON ;FDC-INTERRUPT
37
38 10770 034164 LDA 3,SAVOR
39 10771 030470 LDA 2,INTAL
40 10772 156413 SUBW 2,3,SNR ;SKIP IF NOT LOOP B001
41 10773 000432 JMP INTON ;IF INTA-LOOP THEN GOTO TESTLOOP
42
43 10774 034462 LDA 3,CLINS ;ILLEGAL INTERRUPT
44 10775 030460 LDA 2,ILDEV
45 10776 157000 ADD 2,3 ;FORM CLEAR INSTR.
46 10777 054401 STA 3,.+1
47 11000 000000 U ;CLEAR INSTR. TO ILL. DEV. IS PLACED
48 11001 006043 CCRLF ;HERE
49 11002 006040 CMESS
50 11003 011062 ILINT
51 11004 024451 LDA 1,ILDEV
52 11005 006052 CTOCT
53 11006 014451 DSZ ILICO
54 11007 000406 JMP IPETR
55 11010 034221 LDA 3,FOO05
56 11011 054446 STA 3,ILICO ;RESTORE ILL.INTR. COUNTER
57 11012 006006 RESET
58 11013 002401 JMP R.+1
59 11014 000014A SWISA ;5 ILL. INTR. RECEIVED - GIVE UP

```

10038 .MAIN

```
01
02 11015 034433 IRETR: LDA 3,ISAV3
03 11016 030433 LDA 2,ISAV2
04 11017 024433 LDA 1,ISAV1
05 11020 020434 LDA 0,ISAVC
06 11021 101100 MOVL 0,0
07 11022 020431 LDA 0,ISAVU
08 11023 060177 INTEN
09 11024 002176 JMP 4IRTSV ;RETURN FROM INTERRUPT
10 11025 034427 INTON: LDA 3,ISAVC
11 11026 175100 MOVL 3,3
12 11027 030422 LDA 2,ISAV2
13 11030 034342 LDA 3,INTAD
14 11031 175015 MOVW 3,3,SNR
15 11032 000403 JMP .+3
16 11033 034415 LDA 3,ISAV3
17 11034 002342 JMP 4INTAD ;GO TO INTR. ROUT. INDICATED BY "INTAD"
18 11035 034413 LDA 3,ISAV3
19 11036 002176 JMP 4IRTSV ;RETURN IMMEDIATLY IF "INTAD" = 0
20
21 11037 063077 PWINT: HALT ;POWER INTERRUPT DETECTED.
22
23 11040 020267 RTCRI: LDA 0,MINDR
24 11041 040266 STA 0,CURUN
25 11042 102400 SUB 0,0
26 11043 040345 STA 0,MULMK
27 11044 062677 IORST
28 11045 006061 CWAIT
29 11046 000237 FD3K0
30 11047 002411 JMP 4IFTS5
31
32 11050 000000 ISAV3: 0
33 11051 000000 ISAV2: 0
34 11052 000000 ISAV1: 0
35 11053 000000 ISAV0: 0
36 11054 000000 ISAVC: 0
37 11055 000000 ILDEV: 0
38 11056 060200 CLINS: NI0C 0
39 11057 000005 ILIC0: 5
40 11060 007440 IFTS5: FTS50
41 11061 012715 INTAL: 8001+4
42 11062 146311 ILINT: .TXTE ;ILLEGAL INTERRUPT FROM DEV.: !
43 142714
44 040507
45 120314
46 047311
47 142724
48 151322
49 050125
50 120324
51 151306
52 046717
53 042240
54 053305
55 035056
56 000240
```


!0059 .MAIN

```
01
02 11101 014347 RTCIN: DSZ      SYSCLK
03 11102 000406      JMP      .+6
04 11103 126220      ADCLR   1,1      ;AC1= 077777
05 11104 006165      IDUADM
06 11105 000733      JMP      RTCRI
07 11106 000640      JMP      RSPRO
08 11107 000401      JMP      .+1
09 11110 126520      SUBZL   1,1      ;CLOCK INTERRUPT
10 11111 020273      LDA     0,CLCK0 ;DECREMENT ALL PROCESS-CLOCKS
11 11112 101014      MOVW   0,0,SZR
12 11113 122400      SUB     1,0
13 11114 040273      STA     0,CLCK0
14 11115 020273      LDA     0,CLCK1
15 11116 101014      MOVW   0,0,SZR
16 11117 122400      SUB     1,0
17 11120 040273      STA     0,CLCK1
18 11121 020277      LDA     0,CLCK2
19
20 11122 101014      MOVW   0,0,SZR
21 11123 122400      SUB     1,0
22 11124 040277      STA     0,CLCK2
23 11125 020301      LDA     0,CLCK3
24 11126 101014      MOVW   0,0,SZR
25 11127 122400      SUB     1,0
26 11130 040301      STA     0,CLCK3
27 11131 024447      LDA     1,CURPR ;GET CURRENT PROCESS
28 11132 125414      INCW   1,1,SZR
29 11133 000403      JMP      .+3
30 11134 024267      LDA     1,MINDR
31 11135 000421      JMP      RTCI1+1
32 11136 034444      LDA     3,PSTAB ;PROCESS SAVE TABLE
33 11137 137000      ADD     1,3
34 11140 035400      LDA     3,0,3
35 11141 020712      LDA     0,ISAV0 ;SAVE PROCESS SITUATION
36 11142 041400      STA     0,0,3
37 11143 020707      LDA     0,ISAV1
38 11144 041401      STA     0,1,3
39 11145 020704      LDA     0,ISAV2
40 11146 041402      STA     0,2,3
41 11147 020701      LDA     0,ISAV3
42 11150 041403      STA     0,3,3
43 11151 020703      LDA     0,ISAVC
44 11152 041404      STA     0,4,3
45 11153 020176      LDA     0,IRTSV
46 11154 041405      STA     0,5,3
47 11155 125400 RTCI1: INC     1,1      ;INCREMENT TO NEXT PROCESS
48 11156 020270      LDA     0,MAXDR
49 11157 122433      SUBZ0   1,0,SNC ;SKIP IF AC1 <= MAXDR
50 11160 024267      LDA     1,MINDR
51 11161 044417      STA     1,CURPR ;CHANGE CURRENT PROCESS
52 11162 034420      LDA     3,PSTAB ;LOAD NEW PROCESS SITUATION
53 11163 137000      ADD     1,3
54 11164 035400      LDA     3,0,3
```

10040 MAIN

```
01
02 11165 021400 LDA 0,0,3
03 11166 025401 LDA 1,1,3
04 11167 031404 LDA 2,4,3
05 11170 151100 MOVL 2,2
06 11171 031405 LDA 2,5,3
07 11172 050407 STA 2,CLKRT
08 11173 031402 LDA 2,2,3
09 11174 035403 LDA 3,3,3
10 11175 060114 NIOS RTC ;START CLOCK
11 11176 060177 INTEN
12 11177 002402 JMP 4,CLKRT ;GO TO NEXT PROCESS IN CHAIN
13
14 11200 000000 CURPR: 0 ;CURRENT PROCESS
15 11201 000000 CLKRT: 0
16 11202 011203 PSTAB: .+1
17 11203 011207 PTAB0
18 11204 011215 PTAB1
19 11205 011223 PTAB2
20 11206 011231 PTAB3
21 ;PROCESS SAVE AREAS
22 11207 000000 PTAB0: 0 ;AC0
23 11210 000000 0 ;AC1
24 11211 000000 0 ;AC2
25 11212 000000 0 ;AC3
26 11213 000000 0 ;CARRY
27 11214 000000 0 ;LAST INTERRUPT ADDRESS
28
29 11215 000000 PTAB1: 0
30 11216 000000 0
31 11217 000000 0
32 11220 000000 0
33 11221 000000 0
34 11222 000000 0
35
36 11223 000000 PTAB2: 0
37 11224 000000 0
38 11225 000000 0
39 11226 000000 0
40 11227 000000 0
41 11230 000000 0
42
43 11231 000000 PTAB3: 0
44 11232 000000 0
45 11233 000000 0
46 11234 000000 0
47 11235 000000 0
48 11236 000000 0
```

!0041 .MAIN

```
01
02                                     ;MULTI DRIVE INTERRUPT SERVICE.
03 11237 070401 MDINT:  DIA      2,XFDC ;GET INTR. IDENT. REG.
04 11240 034435      LDA      3,SEMBAS
05 11241 024211      LDA      1,FOU10
06 11242 147415      ANDW    2,1,SNR
07 11243 000404      JMP      DR0NI ;NOT DRIVE 0
08 11244 102440      SUBO    0,0
09 11245 061061      OOA     0,XFDC ;STOP DRIVE 0, CLEAR INTR.
10 11246 043400      STA     0,A0,3 ;OPEN SEMAPHORE
11 11247 125220 DR0NI:  MOVZR   1,1
12 11250 147415      ANDW    2,1,SNR
13 11251 000405      JMP      DR1NI ;NOT DRIVE 1
14 11252 102520      SUBZL   0,0
15 11253 061061      OOA     0,XFDC ;STOP DRIVE 1, CLEAR INTR.
16 11254 102440      SUBO    0,0
17 11255 043401      STA     0,A1,3 ;OPEN SEMAPHORE
18 11256 125220 DR1NI:  MOVZR   1,1
19 11257 147415      ANDW    2,1,SNR
20 11260 000405      JMP      DR2NI ;NOT DRIVE 2
21 11261 020216      LDA     0,F0002
22 11262 061061      OOA     0,XFDC ;STOP DRIVE 2, CLEAR INTR.
23 11263 102440      SUBO    0,0
24 11264 043402      STA     0,A2,3 ;OPEN SEMAPHORE
25 11265 125220 DR2NI:  MOVZR   1,1
26 11266 147415      ANDW    2,1,SNR
27 11267 000405      JMP      DR3NI ;NOT DRIVE 3
28 11270 020217      LDA     0,F0003
29 11271 061061      OOA     0,XFDC ;STOP DRIVE 3, CLEAR INTR.
30 11272 102440      SUBO    0,0
31 11273 043403      STA     0,A3,3 ;OPEN SEMAPHORE
32 11274 002406 DR3NI:  JMP      *COMIR ;RETURN FROM INTERRUPT.
33
34
35 11275 011276 SEMBAS:  .+1
36 11276 000272      SEMA0
37 11277 000274      SEMA1
38 11300 000276      SEMA2
39 11301 000300      SEMA3
40
41 11302 011015 COMIR:  IRETR
```

10042 .MAIN

```
01
02 ;PROCEDURE CALL: RECAL
03 ;
04 ;THE PROCEDURE WILL RECALIBRATE CURRENT UNIT. IT WILL WAIT FOR
05 ;UNIT-DONE AND CHECK THE RECEIVED STATUS.
06 ;
07 ;RETURN: .+1
08 ;
09 ;DESTROYED: ACS
10
11
12 11303 054164 RCALI: STA 3,SAVDR ;SAVE AC'S
13 11304 040165 STA 0,SAV00
14 11305 044166 STA 1,SAV01
15 11306 050167 STA 2,SAV02
16
17 11307 034242 LDA 3,BASAD
18 11310 076061 DOB 3,XFDC ;SET BASE ADDRESS
19 11311 020266 LDA 0,CURUN ;AC0= CURRENT UNIT
20 11312 117000 ADD 0,3
21 11313 035400 LDA 3,0,3 ;AC3= CURRENT COMMAND LOC.
22 11314 024246 LDA 1,RECOM ;AC1= RECALIBRATE COMMAND
23 11315 045400 STA 1,0,3
24 11316 024210 LDA 1,.1B13
25 11317 107000 ADD 0,1 ;ADD START-BIT TO UNITNO.
26 11320 065061 DOA 1,XFDC ;START RECALIBRATION
27
28 11321 006065 TIMSK ;WAIT FOR FDC-DONE (MAX 3.5 SEC)
29 11322 006654 3500.
30 11323 065601 SKPDR XFDC
31 11324 000402 JMP .+2 ;ERROR RETURN
32 11325 000404 JMP .+4
33 11326 030164 LDA 2,SAVDR
34 11327 006162 ERROR ;FDC-DONE ISN'T SET WITHIN 3500 MS
35 ;AFTER START OF RECAL. AC2= CALL OF
36 ;RECAL-PROCEDURE.
37 11330 000421 JMP RCALO
38
39 11331 006143 IDUNIT ;GET INTERRUPT-UNIT TO AC1
40 11332 006162 ERROR ;ILLEGAL INTR.IDENT.REG. READ.
41 ;AC1= REGISTER READ.
42 11333 020266 LDA 0,CURUN
43 11334 106414 SUB# 0,1,SZR
44 11335 006162 ERROR ;INTERRUPT FROM UNEXPECTED UNIT.
45 ;AC1= UNEXPECTED UNIT NO.
46 11336 061061 DOA 0,XFDC ;CLEAR INTERRUPT
47 11337 034242 LDA 3,BASAD
48 11340 117000 ADD 0,3
49 11341 035400 LDA 3,0,3
50 11342 021405 LDA 0,5,3 ;GET STATUS WORD
51 11343 006116 STATN ;CHECK STATUS
52 11344 175777 175777
53 11345 000402 JMP .+2 ;STATUS ERROR
54 11346 000403 JMP RCALO
55 11347 030164 LDA 2,SAVDR
56 11350 006162 ERROR ;STATUS ERROR AFTER RECALIBRATION.
57 ;AC0= STATUS READ, AC1="NON"-EX-
58 ;PECTED STATUS,AC2= CALL OF "RECAL"-
59 ;ROUTINE
60 11351 020165 RCALO: LDA 0,SAV00
61 11352 024166 LDA 1,SAV01
62 11353 030167 LDA 2,SAV02
63 11354 002164 JMP RSAVDR
```

0043 .MAIN

```
01
02 ;PROCEDURE CALL: GRAND
03 ;
04 ;THE ROUTINE WILL GENERATE A RANDOM NUMBER, USING THE VALUE
05 ;IN AC0 AS "OLD" NUMBER. THE RANDOM NO. IS DELIVERED IN AC0.
06 ;
07 ;ENTRY: AC0= LAST USED RANDOM NO.
08 ;
09 ;RETURN: .+1
10 ;
11 ;DESTROYED: AC0,AC3
12
13 11355 054164 GERAN: STA 3,SAVOR
14 11356 044166 STA 1,SAV01
15 11357 050167 STA 2,SAV02
16
17 11360 024422 LDA 1,RANCA
18 11361 044423 STA 1,RANCO ;SET COUNTER
19 11362 105120 MOVZL 0,1
20 11363 125120 MOVZL 1,1
21 11364 014420 DSZ RANCO ;DECREMENT COUNTER
22 11365 000776 JMP .-2
23 11366 107000 ADD 0,1 ;8 TIMES SHIFTED
24 11367 125120 MOVZL 1,1
25 11370 125120 MOVZL 1,1
26 11371 123000 ADD 1,0
27 11372 034411 LDA 3,RANCB
28 11373 163000 ADD 3,0
29 11374 101112 MOVLE 0,0,SZC
30 11375 174220 COMZR 3,3
31 11376 054405 STA 3,RANCB
32 11377 024166 LDA 1,SAV01
33 11400 030167 LDA 2,SAV02
34 11401 002164 JMP *SAVOR
35
36 11402 000010 RANCA: 10
37 11403 033031 RANCB: 33031
38 11404 000000 RANCO: 0
39
40
41
42 ;PROCEDURE CALL: SETIN
43 ;
44 ;A PROCEDURE TO PLACE THE COMMON INTERRUPT SERVICE ROUT. ADDRESS
45 ;IN LOC 1, THE FDC-INTR. SERVICE LOCATION IN LOC "INTAD",
46 ;AND TO MASK OUT ALL DEVICES EXCEPT FDC AND TTI.
47 ;
48 ;ENTRY: AC0= ADDRESS OF FDC INTR. SERVICE ROUTINE.
49 ;
50 ;RETURN: CALL+1
51 ;
52 ;DESTROYED: AC3
53
54 11405 054175 STINT: STA 3,RETU1
55 11406 040342 STA 0,INTAD
56 11407 006006 RESET
57 11410 002175 JMP &RETU1
```

```

10044 .MAIN
01
02 ;PROCEDURE CALL: RANTS
03 ;
04 ;THE ROUTINE WILL, BY USE OF PROCEDURE "GRAND", GENERATE A
05 ;RANDOM TRACK NO. AND A RANDOM SECTOR NO. THE TRACK NO. WILL
06 ;BE DELIVERED IN AC1-R, THE SECTOR NO. IN AC0-L, WITH RANDOM SUR-
07 ;FACE BIT.
08 ;
09 ;RETURN: .+1
10 ;
11 ;DESTROYED: AC0,AC1,AC3
12
13 11411 054170 RTRSE: STA 3,SAV1R
14 11412 050173 STA 2,SAV12
15 11413 020457 LDA 0,LARAN ;LAST USED RANDOM NO.
16 11414 006134 GRAND ;GET NEW RANDOM NO. TO AC0
17 11415 040455 STA 0,LARAN
18 11416 024455 RESEC: LDA 1,FMSK4 ;AC1:= 37
19 11417 107400 AND 0,1 ;AC1:= 0
20 11420 030334 LDA 2,MXSEC ;AC2:= 26.
21 11421 146032 ADCZ0 2,1,SZC ;SKIP IF MAX. SEC. >= 0
22 11422 000405 JMP NWSEC ;GET A NEW NUMBER
23 11423 125005 MOV 1,1,SNR ;SECTOR 0 NOT ALLOWED
24 11424 000403 JMP NWSEC
25 11425 044450 STA 1,ASEC ;SECTOR NO IS OK
26 11426 000405 JMP GETTR ;GET TRACK NO.
27
28 11427 020443 NWSEC: LDA 0,LARAN ;TRY TO GET A NEW 0
29 11430 006134 GRAND
30 11431 040441 STA 0,LARAN
31 11432 000764 JMP RESEC
32
33 11433 020457 GETTR: LDA 0,LARAN ;GET A RANDOM TRACK NO.
34 11434 024440 RETRA: LDA 1,FMSK5 ;AC1:= 007740
35 11435 107400 AND 0,1
36 11436 125220 MOVZR 1,1
37 11437 125220 MOVZR 1,1
38 11440 125220 MOVZR 1,1
39 11441 125220 MOVZR 1,1
40 11442 125220 MOVZR 1,1 ;AC1:= 0
41 11443 030333 LDA 2,MXTRA ;AC2:= 76.
42 11444 146032 ADCZ0 2,1,SZC ;SKIP IF MAX. TRACK >= 0
43 11445 000405 JMP NWTRA ;TRY TO GET A NEW NO.
44 11446 125015 MOV0 1,1,SNR ;TRACK 0 IS SPECIAL
45 11447 000403 JMP NWTRA
46 11450 044426 STA 1,RATRA ;TRACK NO. IS OK
47 11451 000405 JMP RANTO ;RETURN
48
49 11452 020420 NWTRA: LDA 0,LARAN ;TRY TO GET A NEW 0
50 11453 006134 GRAND
51 11454 040416 STA 0,LARAN
52 11455 000757 JMP RETRA
53
54 11456 020417 RANTO: LDA 0,ASEC ;AC0:= SECTOR NO.
55 11457 101300 MOVS 0,0
56 11460 024412 LDA 1,LARAN
57 11461 030200 LDA 2,.1B0
58 11462 147400 AND 2,1
59 11463 034271 LDA 3,MAXSU
60 11464 175015 MOV0 3,3,SNR
61 11465 126440 SUB0 1,1 ;MASK OUT SURFACE BIT IF "MAXSU"=0
62 11466 125000 ADD 1,0 ;ADD SURFACE BIT

```

10045 .MAIN

01

02 11467 024407 LDA 1,RATRA ;AC1:= TRACK NO.

03 11470 030173 LDA 2,SAV12

04 11471 002170 JMP &SAV1R

05

06 11472 012345 LARAN: 12345 ;LAST GENERATED RANDOM NO.

07 11473 000037 FMSK4: 000037

08 11474 007740 FMSK5: 007740

09 11475 000000 RASEC: 0 ;GENERATED SECTOR NO.

10 11476 000000 RATRA: 0 ;GENERATED TRACK NO.

11

12

13

14 ;PROCEDURE CALL: RESET

15

16 11477 054457 RSTIO: STA 3,RSRET

17 11500 040457 STA 0,RSSV0

18 11501 020345 LDA 0,MULMK

19 11502 101014 MOV0 0,0,SZR

20 11503 000410 JMP RSTSK ;SKIP DRIVE STOP IF MULTI DRIVE TEST

21 11504 060277 INTDS

22 11505 102440 SUB0 0,0

23 11506 061061 DOA 0,XFDC ;STOP DRIVE 0

24 11507 101400 INC 0,0

25 11510 061061 DOA 0,XFDC ;STOP DRIVE 1

26 11511 101400 INC 0,0

27 11512 061061 DOA 0,XFDC ;STOP DRIVE 2

28 11513 101400 INC 0,0

29 11514 061061 DOA 0,XFDC ;STOP DRIVE 3

30 11515 063761 SKPDZ XFDC

31 11516 060261 NIOC XFDC

32 11517 034436 LDA 3,RSMASK

33 11520 076077 MSKO 3

34 11521 063511 RSTSK: SKPBZ T10

35 11522 000777 JMP .-1

36 11523 063535 SKPRZ DIS

37 11524 000777 JMP .-1

38 11525 060217 NIOC LPT

39 11526 060211 NIOC T10

40 11527 060235 NIOC DIS

41 11530 034430 LDA 3,ACOMI

42 11531 054001 STA 3,1

43 11532 0340214 LDA 3,POWZE

44 11533 054000 STA 3,0

45 11534 020345 LDA 0,MULMK

46 11535 101014 MOV0 0,0,SZR

47 11536 000406 JMP RSTNI ;IF MULTI DRIVE TEST THE GOTO "RSTNI"

48 11537 102440 SUB0 0,0

49 11540 040342 STA 0,INTAD

50 11541 020410 LDA 0,RSSV0

51 11542 060177 INTEN

52 11543 002413 JMP &RSRET ;RETURN

53

54 11544 020415 RSTNI: LDA 0,ERRTA

55 11545 101400 INC 0,0

56 11546 034410 LDA 3,RSRET

57 11547 116414 SUB0 0,3,SZR ;SKIP IF CALLED FROM "EHALT"-ST.ROUT.

58 11550 000403 JMP .+3

59 11551 020406 LDA 0,RSSV0

60 11552 002412 JMP &MULR

61 11553 020404 LDA 0,RSSV0

62 11554 002402 JMP &RSRET

!0046 .MAIN

```
01
02 11555 177375 RSMASK: 177375
03 11556 000000 RSRET: 0
04 11557 000000 RSSVO: 0
05 11560 010730 ACOMI: INTSV
06 11561 000005AERRTA: ERRET
07 11562 000005AERRRC: ERRCT
08 11563 000000 MERRT: 0
09 11564 000000 MULER: 0 ;RETURN ADDR. IS PLACED HERE BY MERO -
10 ;MERS
11 11565 000000 UMESR: 0
12 11566 000000 UMESU: 0
13
14
15
16 ;PROCEDURE CALL: SWENB
17 ;
18 ;THE PROCEDURE WILL ENABLE THE PANEL-SWITCHES BY MAKING
19 ;"SWREG" = 0. IT'S CALLED ONLY FROM STANDARD ROUTINE "CSAMS".
20 ;
21 ;RETURN: .+1
22 ;
23 ;DESTROYED: AC3
24
25 11567 054774 ENBSW: STA 3,MERRT
26 11570 176440 SUBO 3,3 ;AC3:= 0
27 11571 054132 STA 3,SWREG
28 11572 002771 JMP 4MERRT
29
30
31
32
33
34 ;PROCEDURE TO MESS CURRENT DRIVE WITH ERROR.
35
36 11573 054772 UNMES: STA 3,UMESR ;MESS CURRENT DRIVE
37 11574 050447 STA 2,XESV2
38 11575 044445 STA 1,XESV1
39 11576 040770 STA 0,UMESU
40 11577 060277 INTDS
41 11600 006073 CRESW ;READ SWITCHES TO AC2
42 11601 034206 LDA 3,.1B10
43 11602 157414 ANDO 2,3,SZF ;SKIP IF PRINT ENABLE
44 11603 000417 JMP UNMSO
45 11604 036756 LDA 3,#SEERRC
46 11605 175014 MOVQ 3,3,SZR ;SKIP IF FIRST ERROR IN THIS LOOP
47 11606 000414 JMP UNMSO
48 11607 006046 CDICL
49 11610 006040 CMESR
50 11611 011725 TCURDR
51 11612 006044 CDISP
52 11613 011736 DCURDR
53 11614 024752 LDA 1,UMESU
54 11615 006053 CTDEC
55 11616 024750 LOA 1,UMESU
56 11617 006057 CDEEC
57 11620 006061 CWAIT
58 11621 000251 FD128
59 11622 020417 UNMSO: LDA 0,XESVO
60 11623 024417 LDA 1,XESV1
61 11624 030417 LDA 2,XESV2
62 11625 002740 JMP 4UMESR
```


10047 MAIN

01

02

;PROCEDURE CALL: ERROR

03

;

04

;EXTENSION OF STANDARD PROCEDURE "EHALT"

05

06 11626 054412 XEHALT: STA 3,XERET

07 11627 040412 STA 0,XESV0

08

09 11630 020414 LDA 0,XEAD0 ;ADDRESS OF B003

10 11631 162432 SUBZ0 3,0,SZC ;SKIP IF AC3 > B003

11 11632 000404 JMP XEHA0

12

13 11633 020266 LDA 0,CURUN

14 11634 004737 JSR UNMES

15 11635 054403 LDA 3,XERET

16 11636 020403 XEHA0: LDA 0,XESV0

17 11637 002114 JMP #114 ;EHALT WITHOUT JSR

18

19

;RETURN IS DONE DIRECTLY TO CALL OF

20

;"ERROR"

21

22 11640 000000 XERET: 0

23 11641 000000 XESV0: 0

24 11642 000000 XESV1: 0

25 11643 000000 XESV2: 0

26 11644 012755 XEAD0: B003

27

28

29

;ERROR-ROUTINES FOR USE IN MULTI-PROCESS SYSTEMS

30

31 11645 060277 MER0: INTDS

32 11646 054715 STA 3,MERRT

33 11647 040772 STA 0,XESV0

34 11650 102440 SUB0 0,0

35 11651 004722 JSR UNMES

36 11652 054406 LDA 3,MERUR

37 11653 054711 STA 3,MULER

38 11654 034707 LDA 3,MERRT

39 11655 002114 JMP #114 ;EHALT WITHOUT JSR

40 11656 060177 INTEN

41 11657 002704 JMP #MERRT

42 11660 011656 MER0R: -2

43

44 11661 060277 MER1: INTDS

45 11662 054701 STA 3,MERRT

46 11663 040756 STA 0,XESV0

47 11664 102520 SUBZL 0,0

48 11665 004706 JSR UNMES

49 11666 034406 LDA 3,MER1R

50 11667 054675 STA 3,MULER

51 11670 034673 LDA 3,MERRT

52 11671 002114 JMP #114

53 11672 060177 INTEN

54 11673 002670 JMP #MERRT

55 11674 011672 MER1R: -2

!0048 .MAIN

```
01
02 11675 060277 MER2: INTDS
03 11676 054665 STA 3,MERRT
04 11677 040742 STA 0,XESVU
05 11700 020216 LDA 0,FD002
06 11701 004672 JSR UNMES
07 11702 034406 LDA 3,MER2R
08 11703 054661 STA 3,MULER
09 11704 034657 LDA 3,MERRT
10 11705 002114 JMP R114
11 11706 060177 INTEN
12 11707 002654 JNP 4MERRT
13 11710 011706 MER2R: .-2
14
15 11711 060277 MER3: INTDS
16 11712 054651 STA 3,MERRT
17 11713 040726 STA 0,XESVU
18 11714 020217 LDA 0,FD003
19 11715 004656 JSR UNMES
20 11716 034406 LDA 3,MER3R
21 11717 054645 STA 3,MULER
22 11720 034643 LDA 3,MERRT
23 11721 002114 JMP R114
24 11722 060177 INTEN
25 11723 002640 JNP 4MERRT
26 11724 011722 MER3R: .-2
27
28
29 11725 005215 TCURDR: .TXTE !<15><12>CURRENT DRIVE: !
30 052703
31 151322
32 047305
33 120324
34 151104
35 053311
36 035305
37 000240
38
39 11736 052703 DCURDR: .TXTE !CUR.DRIVE: !
40 027322
41 151104
42 053311
43 035305
44 000000
```

10049 .MAIN

```
01
02 ;PROCEDURE CALL: RDSWS
03 ;
04 ;THE PROCEDURE IS AN EXTENSION OF THE STANDARD PROCEDURE "XRESW"
05 ;WHICH WILL READ AND ANNOUNCE THE SWITCH-SETTING. "RDSWS" WILL
06 ;USE "SWREG" AS SWITCHCONTROL IF IT'S <> 0, ELSE THE PROC. WILL
07 ;READ THE SWITCHES AND CHECK IF ANY OF THE SWITCHES 1-9 ARE SET.
08 ;IF THAT'S THE FACT, THE SETTING OF SWITCH 13 - 15 WILL BE DIS-
09 ;ABLED TO PROHIBIT PROGRAM-HALTS, CAUSED BY THIS ERRONEOUS SWITCH
10 ;SETTING.
11 ;
12 ;RETURN: .+1 IF "INIMK" = 0
13 ; IF "SWREG" <> 0 THEN AC2= "SWREG", ELSE
14 ; IF NONE OF SW 1-9 ARE SET, THEN AC2= SWITCHES READ
15 ; IF ANY OF SW 1-9 ARE SET, THEN AC2= SWITCHES READ
16 ; WITH OUTMASKED BIT 13-15
17 ;
18 ; TO "MSPRO" IF "INIMK" <> 0.
19 ;
20 ;DESTROYED: AC3, AC2 (AND AC1 IF "SWREG" <> 0)
21
22 11744 054427 RDSWI: STA 3, SWRET
23 11745 040427 STA 0, SWSAV
24
25 11746 030132 LDA 2, SWREG ;AC2:= COMMON SWITCH REGISTER
26 11747 151014 MOVW 2, 2, SZR ;SKIP IF REG = 0
27 11750 145000 MOV 2, 1 ;THIS WILL FORCE "UNCHANGED"? YES.
28 11751 034131 LDA 3, INIMK ;AC3 <> 0 IF FIRST CALL AFTER LOAD
29 11752 175014 MOVW 3, 3, SZR ;SKIP IF NOT FIRST CALL AFTER PROG. LOAD
30 11753 000411 JMP INI01
31
32 11754 020415 LDA 0, SWMS1 ;AC0:= 077700
33 11755 034415 LDA 3, SWMS2 ;AC3:= 177770
34 11756 151015 MOVW 2, 2, SNR ;SKIP IF SWITCH REG. <> 0
35
36 11757 070477 READS 2
37 11760 143414 ANDW 2, 0, SZR
38 11761 173400 AND 3, 2 ;MASK OUT SW 13-15
39 11762 020412 LDA 0, SWSAV
40 11763 002410 JMP #SWRET ;RETURN TO PROC. "XRESW"
41
42 11764 176440 INI01: SUBO 3, 3
43 11765 054131 STA 3, INIMK ;CLEAR INI-MARK
44 11766 145000 MOV 2, 1
45 11767 002401 JMP #IMSPR ;GO TO "REBIN" AND MESS PROG. NAME
46
47 11770 000013#IMSPR: MSPRO
48 11771 077700 SWMS1: 077700
49 11772 177770 SWMS2: 177770
50 11773 000000 SWRET: 0
51 11774 000000 SWSAV: 0
```

!0050 .MAIN

```
01
02 ;PROCEDURE CALL: COMPT
03 ;
04 ;THE ROUTINE WILL CHANGE EVERY BYTE IN THE WRITE BUFFER TO IT'S
05 ;COMPLEMENT.
06 ;
07 ;RETURN: .+1
08 ;
09 ;DESTROYED: ACS
10
11 11775 054164 PATCM: STA 3,SAVOR
12 11776 044166 STA 1,SAV01
13 11777 050167 STA 2,SAV02
14
15 12000 024425 LDA 1,PATT1 ;CHANGE PATTERNWORD
16 12001 124000 COM 1,1
17 12002 044425 STA 1,PATT1
18 12003 000416 JMP PATCO
19
20
21 ;PROCEDURE CALL: PTFLL
22 ;
23 ;THE ROUTINE WILL FILL THE WRITE BUFFER (WRBUF) WITH 128 BYTES
24 ;TAKEN ALTERNATELY FROM PATT1 AND PAT12.
25 ;
26 ;RETURN: .+1
27 ;
28 ;DESTROYED: ACS
29
30 12004 054164 PATFL: STA 3,SAVOR
31 12005 044166 STA 1,SAV01
32 12006 050167 STA 2,SAV02
33
34 12007 024535 PATFC: LDA 1,SECSIZ
35 12010 125220 MOVZR 1,1
36 12011 044415 STA 1,CHACP ;INITIALIZE CHARCOUNTER
37 12012 030250 LDA 2,AWBUF ;AC2:= ADDRESS OF WRITE BUFFER
38
39 12013 024412 LDA 1,PATT1
40 12014 045000 PASWB: STA 1,0,2
41 12015 151400 INC 2,2
42 12016 124000 COM 1,1
43 12017 014405 DSZ CHACP
44 12020 000774 JMP PASWB ;MOVE MORE PATTERNS
45
46 12021 024166 PATCO: LDA 1,SAV01 ;WRITE BUFFER IS FULL - EXIT
47 12022 030167 LDA 2,SAV02
48 12023 002164 JMP RSAVOR
49
50 12024 000000 CHACP: 0 ;CHARCOUNTER
51 12025 125125 PATT1: 125125 ;PATTERN
```

10051 .MAIN

```
01
02 ;PROCEDURE CALL: CALTR
03 ;
04 ;THE PROCEDURE "TRCAL" WILL CALCULATE A NEW TRACK NO. TO AC1
05 ;DEPENDING ON THE PREVIOUS TRACK NO. (IN AC2 AT ENTRY).THE
06 ;CALCULATION IS DONE IN A MANNER TO PRODUCE WORST CASE HEAD-
07 ;MOVEMENT, AND GOES LIKE THIS:
08 ;
09 ; STARTING WITH 0
10 ; NEXT: 76,1,75,2,74,3,73,4,.....36,40,37,39,38.
11 ;
12 ;ENTRY: AC2= LAST USED TRACK NO.
13 ;
14 ;RETURN: +1 IF AC2 AT ENTRY <> 38, WITH AC1= NEW TRACK NO
15 ;
16 ; +2 IF AC2 AT ENTRY = 38, NO NEW NO. IS CALCUL.
17 ;
18 ;DESTROYED: AC1,AC3
19
20
21 12026 054164 TRCAL: STA 3,SAVOR
22
23 12027 034225 LDA 3,F0038 ;AC3:= 38.
24 12030 172415 SUB# 3,2,SNR ;SKIP IF OLD TRACK NO <> 38
25 12031 000416 JMP TCOU2 ;INCREMENT RETURN AND EXIT
26 12032 156033 ADCZ# 2,3,SNR ;SKIP IF OLD TRACK NO < 38.
27
28 12033 000404 JMP TRCA1 ;OLD TRACK NO > 38.
29
30 12034 024335 LDA 1,MXTRA ;OLD TRACK NO < 38., AC1:= 76.
31 12035 146400 SUB 2,1 ;AC1:= 76 - AC2
32 12036 000412 JMP TCOU1 ;RETURN
33
34 12037 175400 TRCA1: INC 3,3 ;AC3:= 39.
35 12040 156414 SUB# 2,3,SNR ;SKIP IF OLD TRACK NO = 39.
36
37 12041 000403 JMP TRCA2 ;OLD TRACK NO > 39.
38
39 12042 024225 LDA 1,F0038 ;OLD TRACK NO = 39., AC1:= 38.
40 12043 000405 JMP TCOU1 ;RETURN
41
42 12044 024335 TRCA2: LDA 1,MXTRA ;AC1:= 76.
43 12045 125400 INC 1,1
44 12046 146401 SUB 2,1,SKP ;AC1:= 77. - AC2
45
46 12047 010164 TCOU2: ISZ SAVOR ;INCREMENT RETURN ADDRESS
47 12050 002164 TCOU1: JMP ASAVOR
```

10052 .MAIN

```
01
02 ;PROCEDURE CALL: WGSTATUS
03 ;
04 ;PROCEDURE WILL GET THE FDC-STATUS FROM CURRENT DEVICE
05 ;TO AC0 BY READING THE STATUS WORD IN CHANNEL PROG. THE STA-
06 ;TUS WILL BE READ WHEN THE WANTED DRIVE SETS DONE.
07 ;
08 ;CALL: GSTATUS
09 ; "ERROR RETURN" (TIME OUT OR UNWANTED DRIVE INTERRUPT)
10 ; "RETURN"
11 ;
12 ;DESTROYED: AC0,AC3
13
14 12051 175400 RSTAT: INC 3,3
15 12052 054164 STA 3,SAV0R
16 12053 044166 STA 1,SAV01
17 12054 050167 STA 2,SAV02
18
19 12055 006063 TIMSK ;WAIT FOR FDC-DONE (MAX 3500 MS)
20 12056 006654 3500.
21 12057 063661 SKPDN XFDC
22 12060 000410 JMP STTIM ;TIME OUT - NO STATUS DELIVERED
23
24 12061 006143 IDUNIT ;GET UNIT NO. TO AC1
25 12062 006162 ERROR ;ILLEGAL INTR.IDENT REG. READ
26 ;AC1= REG. READ
27 12063 020266 LDA 0,CURUN
28 12064 106415 SUB# 0,1,SNR
29 12065 000404 JMP SPECI ;STATUS FROM CUR.UNIT IS RECEIVED
30 12066 050164 LDA 2,SAV0R
31 12067 006162 ERROR ;INTERRUPT FROM UNWANTED DRIVE
32 ;AC0= CURRENT DRIVE,AC1= ERROR DRIVE
33 ;AC2= CALL OF GSTATUS
34 12070 014164 STTIM: DSZ SAV0R
35 12071 034242 SREGI: LDA 3,BASAD
36 12072 117000 ADD 0,3 ;ADD DRIVE NO. TO BASE ADDRESS
37 12073 035400 LDA 3,0,3
38 12074 021405 LDA 0,5,3 ;GET STATUS
39 12075 065061 DOA 1,XFDC ;CLEAR INTERRUPT BY STOP COMMAND
40 12076 063661 SKPDN XFDC ;DONE SHOULD BE CLEARED
41 12077 060177 INTEN
42 12100 024166 LDA 1,SAV01
43 12101 050167 LDA 2,SAV02
44 12102 002164 JMP #SAV0R ;RETURN
```

!0053 .MAIN

```
01
02 ;PROCEDURE CALL: SETDO
03 ;
04 ;THE ROUTINE WILL SET FDC-DONE BY GIVING THE CONTROLLER AN
05 ;ILLEGAL COMMAND.
06 ;
07 ;RETURN: CALL+1 IF FDC-DONE ISN'T SET WITHIN 1 SEC.
08 ; CALL+2 IF DONE IS SET.
09 ;
10 ;DESTROYED: AC3
11
12 12103 054164 DOSET: STA 3,SAVOR
13 12104 040165 STA 0,SAV00
14 12105 044166 STA 1,SAV01
15 12106 050167 STA 2,SAV02
16 12107 034242 LDA 3,BASAD
17 12110 076061 DOB 3,XFDC ;SET BASE ADDRESS
18 12111 020266 LDA 0,CURUN ;AC0= UNIT NO.
19 12112 117000 ADD 0,3
20 12113 035400 LDA 3,0,3 ;AC3= CURRENT COMMAND LOC.
21 12114 126440 SUBO 1,1 ;AC1= 0 = ILLEGAL COMMAND
22 12115 045400 STA 1,0,3
23 12116 045405 STA 1,5,3 ;INITIALIZE STATUSWORD
24 12117 024210 LDA 1,,1B13
25 12120 107000 ADD 0,1
26 12121 065061 DOA 1,XFDC ;START ILLEGAL COMMAND
27
28 12122 006065 TIMSK ;WAIT FOR FDC-DONE (MAX 1 SEC)
29 12123 001750 1000.
30 12124 065661 SKPDN XFDC
31 12125 000402 JMP DOSOU ;ERROR
32
33 12126 010164 ISZ SAVOR ;OK RETURN +2
34 12127 020165 DOSOU: LDA 0,SAV00
35 12130 024166 LDA 1,SAV01
36 12131 030167 LDA 2,SAV02
37 12132 002164 JMP #SAVOR
38
39
40
41 ;PROCEDURE CALL: SETBREG
42 ;
43 ;THE ROUTINE WILL SET FDC-BASE REGISTER (ACCORDING TO CURRENT
44 ;DRIVE), INITIALIZE STATUSWORD TO -1.
45 ;
46 ;RETURN: CALL+1 AC1= CURRENT UNIT, AC3= FIRST ADDR. OF
47 ; CHANNEL PROGRAM.
48 ;
49 ;DESTROYED: AC1,AC3
50
51 12133 054164 SBREG: STA 3,SAVOR
52 12134 034242 LDA 3,BASAD
53 12135 076061 DOB 3,XFDC ;SET BASE REG.
54 12136 024266 LDA 1,CURUN
55 12137 137000 ADD 1,3
56 12140 035400 LDA 3,0,3 ;GET 1. CH.WORD ADDR.
57 12141 126000 ADC 1,1
58 12142 045405 STA 1,5,3 ;INITIALIZE STATUS WORD
59 12143 024262 LDA 1,RBFADR
60 12144 044261 STA 1,CURBF
61 12145 024266 LDA 1,CURUN
62 12146 002164 JMP #SAVOR
```

10054 .MAIN

```
01
02 ;PROCEDURE CALL: IDUNIT
03 ;
04 ;THE PROCEDURE WILL READ THE INTERRUPT IDENT. REGISTER AND DE-
05 ;LIVER THE UNIT NUMBER IN AC1. (0 <= UNITNO. <=3). IN CASE OF
06 ;FAULT AN ERRORRETURN IS PERFORMED.
07 ;
08 ;RETURN:          CALL+1 IF ERROR IN IDENT.REG., AC1= REG.READ
09 ;                CALL+2 IF UNIT IDENTIFIED, AC1=UNIT NO.
10 ;
11 ;DESTROYED:      AC1,AC3
12
13 12147 040171 IDUNIT: STA 0,SAV10
14 12150 050173      STA 2,SAV12
15
16 12151 126440      SUBO 1,1 ;AC1:= 0
17 12152 060461      DIA 0,XFDC ;GET INTR. IDENT REG. - CLEAR INTERRUPT
18 12153 030211      LDA 2,FO010
19 12154 113414      ANDO 0,2,SZR
20 12155 000415      JMP IUOUT ;UNIT 0
21 12156 125400      INC 1,1
22 12157 151220      MOVZR 2,2
23 12160 113414      ANDO 0,2,SZR
24 12161 000411      JMP IUOUT ;UNIT 1
25 12162 125400      INC 1,1
26 12163 151220      MOVZR 2,2
27 12164 113414      ANDO 0,2,SZR
28 12165 000405      JMP IUOUT ;UNIT 2
29 12166 125400      INC 1,1
30 12167 151220      MOVZR 2,2
31 12170 113415      ANDO 0,2,SNR ;SKIP IF UNIT 3
32 12171 105001      MOV 0,1,SKP ;ILLEGAL INT.IDENT.REG.READ
33 12172 175400 IUOUT: INC 3,3
34 12173 063477      SKPRN CPU ;IF INTERRUPT IS OFF, THEN CHECK FOR
35 12174 000402      JMP .+2 ;TELETYPE INTERRUPT.
36 12175 000404      JNP .+4
37 12176 060177      INTEN
38 12177 000401      JMP .+1
39 12200 060277      INTDS
40 12201 020171      LDA 0,SAV10
41 12202 030173      LDA 2,SAV12
42 12203 001400      JMP 0,3 ;RETURN
43
```


!0055 .MAIN

```
01
02 ;PROCEDURE CALL: INIACH
03 ;
04 ;PROCEDURE TO INITIALIZE THE CHANNEL PROGRAMS. THAT MEANS, IT
05 ;WILL CLEAR THE STATUS WORDS, ALL SECTOR- AND TRACK ADDRESSES
06 ;THE NO. OF BYTES, RE-STORE THE DATABUFFER ADDRESS AND FILL THE
07 ;DATABUFFERS WITH DUMMY PATTERNS.
08 ;
09 ;RETURN:          CALL+1
10 ;
11 ;DESTROYED:      AC3
12
13 12204 054164 ICHCOM: STA    3,SAVOR
14 12205 050167          STA    2,SAV02
15 12206 044166          STA    1,SAV01
16 12207 040165          STA    0,SAV00
17
18 12210 020220          LDA    0,FD004
19 12211 040455          STA    0,CLCC0
20 12212 034242          LDA    3,BASAD
21 12213 054454          STA    3,CLBPO
22 12214 034262          LDA    3,RBFADR
23 12215 054453          STA    3,CLRPO
24 12216 034263          LDA    3,WBFADR
25 12217 054452          STA    3,CLWPO
26
27 12220 034447 NXCHC:  LDA    3,CLBPO
28 12221 035400          LDA    3,0,3
29 12222 102440          SUBO   0,0
30 12223 041400          STA    0,0,3 ;CLEAR CHANNEL PROGRAM
31 12224 041401          STA    0,1,3
32 12225 041402          STA    0,2,3
33 12226 041403          STA    0,3,3
34 12227 102000          ADC    0,0
35 12230 041405          STA    0,5,3 ;INITIALIZE STATUS WORD
36
37 12231 030437          LDA    2,CLRPO
38 12232 031000          LDA    2,0,2
39 12233 051404          STA    2,4,3 ;INSERT DATA BUFFER ADDRESS
40
41 12234 024232          LDA    1,FD256 ;INSERT DUMMY BYTES INTO DATABUFFER
42 12235 124400          NEG    1,1
43 12236 020265          LDA    0,DUPAT
44 12237 041000          STA    0,0,2
45 12240 151400          INC    2,2
46 12241 125404          INC    1,1,SZR
47 12242 000775          JMP    .-3
48 12243 030426          LDA    2,CLWPO
49 12244 031000          LDA    2,0,2
50 12245 024232          LDA    1,FD256
51 12246 124400          NEG    1,1
52 12247 020265          LDA    0,DUPAT
53 12250 100000          COM    0,0
54 12251 041000          STA    0,0,2
55 12252 151400          INC    2,2
56 12253 125404          INC    1,1,SZR
57 12254 000775          JMP    .-3
```

10056 .MAIN

```
01
02 12255 010412      ISZ      CLBPO
03 12256 010412      ISZ      CLRPO      ;INCRMENT TO NEXT DRIVE-COM. AREA
04 12257 010412      ISZ      CLWPO
05 12260 014406      DSZ      CLCCO
06 12261 000737      JMP      NXCHC
07
08 12262 020165      LDA      0,SAV00
09 12263 024166      LDA      1,SAV01
10 12264 030167      LDA      2,SAV02
11 12265 002154      JMP      &SAV0R
12
13 12266 000000 CLCCO:  0
14 12267 000000 CLBPO:  0
15 12270 000000 CLRPO:  0
16 12271 000000 CLWPO:  0
17
18
19
20      ;PROCEDURE CALL: INISCH
21      ;
22      ;SIMILAR TO PROC. INIACH, WITH THE EXCEPTION, THAT ONLY ONE CHAN-
23      ;NEL PROGRAM IS CLEARED.
24      ;
25      ;ENTRY:          ACU= CHANNEL PROG. TO BE CLEARED (0,1,2,3)
26
27 12272 054164 ISCH:  STA      3,SAV0R
28 12273 050167      STA      2,SAV02
29 12274 044166      STA      1,SAV01
30 12275 040165      STA      0,SAV00
31
32 12276 126520      SUBZL   1,1
33 12277 044767      STA      1,CLCCO
34
35 12300 054242      LDA      3,BASAD
36 12301 117000      ADD      0,3
37 12302 054765      STA      3,CLBPO
38 12303 034262      LDA      3,RBFADR
39 12304 117000      ADD      0,3
40 12305 054763      STA      3,CLRPO
41 12306 034263      LDA      3,WBFADR
42 12307 117000      ADD      0,3
43 12310 054761      STA      3,CLWPO
44
45 12311 000707      JMP      NXCHC
```

!0057 .MAIN

```
01
02 ;PROCEDURE CALL: EXECUTE
03 ;
04 ;SET AND START CHANNEL PROGRAM TO CURRENT DRIVE. WAIT FOR TERMI-
05 ;NATION. DEFAULT IS: SECTORS PER TRACK = 26.
06 ;
07 ;ENTRY:          AC0= SURFACE, SECTOR ADDR, CYLINDER ADDR.
08 ;              AC1= NO. OF SECTORS
09 ;              AC2= NO. OF BYTES.
10 ;
11 ;RETURN:        CALL+1  TIME OUT
12 ;              CALL+2  OK - AC0= STATUS RECEIVED
13 ;
14 ;DESTROYED:     AC0,AC2
15
16
17 12312 054170  SCHA0: STA      3,SAV1R
18 12313 040171          STA      0,SAV10
19 12314 044172          STA      1,SAV11
20 12315 050173          STA      2,SAV12
21 12316 010170          ISZ      SAV1R
22 12317 034242          LDA      3,BASAD ;SET UP CHANNEL PROG.
23 12320 024266          LDA      1,CURUN
24 12321 137000          ADD      1,3
25 12322 035400          LDA      3,0,3
26 12323 024214          LDA      1,F0377
27 12324 123404          AND      1,0,SZR ;AC0= CYL. ADDR.
28 12325 000403          JMP      .+3
29 12326 024224          LDA      1,F0026 ;IF CYL. 0 THEN SEC. PER TR.= 26
30 12327 000402          JMP      .+2
31 12330 024334          LDA      1,MXSEC
32 12331 125300          MOVS     1,1
33 12332 123000          ADD      1,0
34 12333 041401          STA      0,1,3 ;SET SEC. PER TRACK, CYL. ADDR.
35 12334 020171          LDA      0,SAV10
36 12335 024214          LDA      1,F0377
37 12336 125300          MOVS     1,1
38 12337 123400          AND      1,0
39 12340 024172          LDA      1,SAV11
40 12341 123000          ADD      1,0
41 12342 041402          STA      0,2,3 ;SET SURFACE, SEC. ADDR., NO. OF SEC.
42 12343 051403          STA      2,3,3 ;SET NO. OF BYTES
43 12344 030261          LDA      2,CURBF
44 12345 024266          LDA      1,CURUN
45 12346 133000          ADD      1,2
46 12347 031000          LDA      2,0,2
47 12350 051404          STA      2,4,3 ;SET DATA BUFFER ADDR.= CUR. READBUFFER
48 12351 102000          ADC      0,0
49 12352 041405          STA      0,5,3 ;INITIALIZE STATUS WORD
50 12353 020210          LDA      0,,1B13
51 12354 107000          ADD      0,1
52 12355 065061          DOA      1,XFDC ;START DRIVE
53
54 12356 006145          WGSTATUS ;WAIT FOR DONE AND GET STATUS TO AC0
55 12357 014170          DSZ      SAV1R ;TIME OUT
56 12360 024172          LDA      1,SAV11
57 12361 030173          LDA      2,SAV12
58 12362 002170          JMP      ASAV1R
```

10058 .MAIN

```
01
02 ;THE LOOP ON THIS PAGE MAY BE USED TO BUILD UP YOUR OWN FDC-TEST.
03 ;
04 ;BEFORE START IN LOC. 403 YOU HAS TO DO FOLLOWING:
05 ;
06 ; INSERT WANTED UNIT NO. IN "LBUNI"
07 ; INSERT WANTED COMMAND IN CORRESPONDING CHANNELPROG.
08 ; ON NEXT PAGE.
09 ; INSERT SECTORS PER TRACK, CYLINDER ADDRESS, SECTOR ADDRESS
10 ; NUMBER OF SECTORS, NUMBER OF BYTES IF NECESSARY.
11 ;
12 ;AFTER START THE COMMAND IS EXECUTED, IF DONE ISN'T SET WITHIN 1
13 ;SEC, AN ERRORHALT IS PERFORMED, THE STATUS WORD IS CHECKED, AND
14 ;IF <> 0 AN OTHER ERROHALT IS PERFORMED.
15 ;
16 ;AS MENTIONED IN THE LOOP DIFFERENT INSTRUCTIONS MAY BE CHANGED
17 ;TO LET THE LOOP RUN CONTINUOUSLY, EVEN IF ERRORS OCCOURES.
18
19 12363 054177 LBSTR: STA 3,GRRET ;START OF LOOP BUILDER
20 12364 006110 SETPO
21
22 12365 054242 RELOP: LDA 3,BASAD
23 12366 076061 DDB 3,XFDC ;SET BASE ADDRESS
24 12367 020505 LDA 0,LBUNI ;SAVE UNIT NO AND CHANNEL PROG.
25 12370 040266 STA 0,CURUN
26 12371 117000 ADD 0,3
27 12372 024241 LDA 1,FDMQ5
28 12373 030502 LDA 2,LBCSV
29 12374 035400 LDA 3,0,3
30 12375 021400 LDA 0,0,3
31 12376 041000 STA 0,0,2
32 12377 151400 INC 2,2
33 12400 175400 INC 3,3
34 12401 125404 INC 1,1,SZR
35 12402 000773 JMP .-5
36
37 12403 034262 LDA 3,RBFADR ;FILL READ BUFFER WITH ZERO'S
38 12404 024470 LDA 1,LBUNI
39 12405 137000 ADD 1,3
40 12406 035400 LDA 3,0,3
41 12407 024232 LDA 1,FD256
42 12410 124400 NEG 1,1
43 12411 102440 SUBO 0,0
44 12412 041400 STA 0,0,3
45 12413 175400 INC 3,3
46 12414 125404 INC 1,1,SZR
47 12415 000775 JMP .-3
48
49 12416 034263 LDA 3,WBFADR ;FILL WRITE BUFFER WITH COUNTING PATTERN
50 12417 024455 LDA 1,LBUNI
51 12420 137000 ADD 1,3
52 12421 035400 LDA 3,0,3
53 12422 030232 LDA 2,FD256
54 12423 150400 NEG 2,2
55 12424 102440 SUBO 0,0
56 12425 105400 INC 0,1
57 12426 101300 MOVS 0,0
58 12427 123000 ADD 1,0
59 12430 041400 STA 0,0,3
60 12431 175400 INC 3,3
61 12432 121400 INC 1,0
62 12433 151404 INC 2,2,SZR
63 12434 000771 JMP .-7
```

```

0059 .MAIN
01
02 12435 020437 LDA 0,LBUNI ;LOAD UNIT NO.
03 12436 024210 LDA 1,-1B13
04 12437 107000 ADD 0,1 ;ADD START BIT TO UNIT NO.
05
06 12440 065061 DDA 1,XFDC ;START FDC
07
08 12441 006063 TIMSK ;WAIT FOR DONE (MAX 2 SEC'S)
09 12442 006654 3500.
10 12443 065661 SKPDN XFDC
11 12444 006162 ERROR ;*** MAY BE CHANGED TO JMP .+1 (000401)
12 ;FDC-DONE ISN'T SET WITHIN 3.5 SEC'S
13
14 12445 034242 LDA 3,BASAD
15 12446 020426 LDA 0,LBUNI ;GET UNIT NO.
16 12447 117000 ADD 0,3
17 12450 035400 LDA 3,0,3
18 12451 021405 LDA 0,5,3 ;GET DELIVERED STATUS
19
20 12452 101014 MOVW 0,0,SZR ;SKIP IF STATUS = 0
21 12453 006162 ERROR ;*** MAY BE CHANGED TO JMP .+1 (000401)
22 ;FDC-STATUS <> 0
23
24 12454 063077 HALT ;*** MAY BE CHANGED TO JMP .+1 (000401)
25
26 12455 024241 LDA 1,FDM05
27 12456 030417 LDA 2,LBCSV
28 12457 034242 LDA 3,BASAD
29 12460 020414 LDA 0,LBUNI
30 12461 117000 ADD 0,3
31 12462 035400 LDA 3,0,3
32 12463 021000 LDA 0,0,2
33 12464 041400 STA 0,0,3
34 12465 151400 INC 2,2
35 12466 175400 INC 3,3
36 12467 125404 INC 1,1,SZR
37 12470 000773 JMP -5
38 12471 000674 JMP RELOP ;REPEAT LOOP
39
40 12472 006113 LOOP
41 12473 063077 HALT
42
43
44 12474 000000 LBUNI: 0 ;WANTED UNIT NO. (0,1,2,3)
45
46 12475 012476 LBCSV: .+1
47 12476 000000 0
48 12477 000000 0
49 12500 000000 0
50 12501 000000 0
51 12502 000000 0

```

!0060 .MAIN

01
02 ;***** FDC COMMUNICATION AREAS *****

03
04 12503 012507 CPST0: CPRG0
05 12504 012515 CPST1: CPRG1
06 12505 012523 CPST2: CPRG2
07 12506 012531 CPST3: CPRG3

08
09
10 ;CHANNELPROGRAM FOR DRIVE 0

11
12 12507 000000 CPRG0: 0 ;COMMAND
13 12510 000000 0
14 12511 000000 0
15 12512 000000 0
16 12513 000000 0 ;DATA BUFFER
17 12514 000000 0 ;STATUS

18
19
20 ;CHANNELPROGRAM FOR DRIVE 1

21
22 12515 000000 CPRG1: 0 ;COMMAND
23 12516 000000 0
24 12517 000000 0
25 12520 000000 0
26 12521 000000 0 ;DATA BUFFER
27 12522 000000 0 ;STATUS

28
29
30 ;CHANNELPROGRAM FOR DRIVE 2

31
32 12523 000000 CPRG2: 0 ;COMMAND
33 12524 000000 0
34 12525 000000 0
35 12526 000000 0
36 12527 000000 0 ;DATA BUFFER
37 12530 000000 0 ;STATUS

38
39
40 ;CHANNELPROGRAM FOR DRIVE 3

41
42 12531 000000 CPRG3: 0 ;COMMAND
43 12532 000000 0
44 12533 000000 0
45 12534 000000 0
46 12535 000000 0 ;DATA BUFFER
47 12536 000000 0 ;STATUS

!0061 .MAIN

01

02

;***** FILENAME: FDC13

03

04

;TEST OF BUS,CPU-SKIP,CPU I/O-LOGIC.

05

06

07 12537 054177 GROUP: STA 3,GRRET

08

09 12540 000401 A000: JMP .+1

10 12541 006112 SETP2

11 12542 063500 SKPRZ 0

12 12543 006162 ERROR ;THE SELB-LINE IS GROUNDED

13 12544 006113 LOOP

14

15 12545 000401 A001: JMP .+1

16 12546 006112 SETP2

17 12547 063700 SKPDZ 0

18 12550 006162 ERROR ;THE SELD-LINE IS GROUNDED

19 12551 006113 LOOP

20

21 12552 000401 A002: JMP .+1

22 12553 006112 SETP2

23 12554 102000 ADC 0,0

24 12555 060400 DIA 0,0

25 12556 100015 COM0 0,0,SNR

26 12557 006162 ERROR ;DIA DIDN'T READ ANYTHING (CPU ERROR)

27 12560 006113 LOOP

28

29 12561 000401 A003: JMP .+1

30 12562 006112 SETP2

31 12563 102440 SUB0 0,0

32 12564 060400 DIA 0,0

33 12565 101004 MOV 0,0,SRZ

34 12566 006162 ERROR ;ONE OR MORE DATALINES ARE GROUNDED,

35 ;A ONE IN ACO SHOES WHICH ONE.

36 12567 006113 LOOP

37

38 12570 000401 A004: JMP .+1

39 12571 006112 SETP2

40 12572 102000 ADC 0,0

41 12573 061400 DIB 0,0

42 12574 100015 COM0 0,0,SNR

43 12575 006162 ERROR ;DIB DIDN'T READ ANYTHING (CPU ERROR)

44 12576 006113 LOOP

45

46 12577 000401 A005: JMP .+1

47 12600 006112 SETP2

48 12601 102000 ADC 0,0

49 12602 062400 DIC 0,0

50 12603 100015 COM0 0,0,SNR

51 12604 006162 ERROR ;DIC DIDN'T READ ANYTHING (CPU ERROR)

52 12605 006113 LOOP

53

54 12606 000401 A006: JMP .+1

55 12607 006112 SETP2

56 12610 060261 NI0C XFDC

57 12611 063761 SKPDZ XFDC

58 12612 006162 ERROR ;CLEAR-PULSE FAILED TO CLEAR FDC-DONE

59 12613 006113 LOOP

!0062 .MAIN

```
01
02 12614 000401 A007:  JMP      .+1
03 12615 006112      SETP2
04 12616 062677      IORST
05 12617 063761      SKPDZ      XFDC
06 12620 006162      ERROR      ;IORST FAILED TO CLEAR FDC-DONE
07 12621 006113      LOOP
08
09 12622 000401 A008:  JMP      .+1
10 12623 006112      SETP2
11 12624 062677      IORST
12 12625 063661      SKPDN      XFDC
13 12626 000402      JMP      .+2
14 12627 006162      ERROR      ;CPU-SKIP ERROR
15 12630 006113      LOOP
16
17 12631 000401 A009:  JMP      .+1
18 12632 006112      SETP2
19 12633 060177      INTEN
20 12634 000401      JMP      .+1
21 12635 063477      SKPBN      CPU
22 12636 006162      ERROR      ;INTERRUPT ON (CPU BUSY) SHOULD BE
23 12637 060277      INTDS      ;SET AFTER INTEN-INSTR.
24 12640 006113      LOOP
25
26 12641 000401 A010:  JMP      .+1
27 12642 006112      SETP2
28 12643 060177      INTEN
29 12644 000401      JMP      .+1
30 12645 060277      INTDS
31 12646 063477      SKPBN      CPU
32 12647 000402      JMP      .+2
33 12650 006162      ERROR      ;INTERRUPT ON (CPU BUSY) SHOULD BE
34 12651 006113      LOOP      ;RESET AFTER INTDS-INSTR.
35
36
37 12652 006061      CWAIT      ;IORST WAIT
38 12653 000237      F03K0
39
40
41 12654 002177      JMP      &GRRET
```


!0063 .MAIN

01

02

;TEST OF FDC-DONE (SET AND CLEAR)

03

04 12655 054177 8001: STA 3,GRRET

05

06 12656 000401 8000: JMP .+1

07 12657 006110 SETPO

08 12660 062677 IORST

09 12661 063761 SKPDZ XFDC

10 12662 006162 ERROR ;FDC-DONE ISN'T CLEARED BY IORST

11 12663 006061 CWAIT ;IORST WAIT

12 12664 000237 F03K0

13 12665 034242 LDA 3,BASAD

14 12666 076061 DOB 3,XFDC ;SET BASE ADDRESS

15 12667 020266 LDA 0,CURUN

16 12670 117000 ADD 0,3

17 12671 035400 LDA 3,0,3

18 12672 126440 SUBO 1,1 ;AC1= 0 = ILLEGAL COMMAND

19 12673 045400 STA 1,0,3 ;SET COMMAND

20 12674 024210 LDA 1,.1B13

21 12675 107000 ADD 0,1 ;ADD START BIT TO UNIT.NO.

22 12676 065061 DOA 1,XFDC ;START FDC

23 12677 006063 TIMSK ;WAIT FOR FDC-DONE (MAX 1 SEC)

24 12700 001750 1000.

25 12701 063661 SKPDN XFDC

26 12702 006162 ERROR ;FDC-DONE ISN'T SET WITHIN 1 SEC.

27 ;AFTER START OF AN ILLEGAL COMMAND.

28 12703 060261 NIOC XFDC

29 12704 063761 SKPDZ XFDC

30 12705 006162 ERROR ;IT'S IMPOSSIBLE TO CLEAR FDC-DONE

31 ;BY USE OF A CLEAR-PULSE.

32 12706 006061 CWAIT ;IORST WAIT

33 12707 000237 F03K0

34 12710 006113 LOOP

35

36

;TEST OF FDC-INTERRUPT

37

38 12711 000401 8001: JMP .+1

39 12712 006112 SETP2

40 12713 060177 INTEN

41 12714 006144 SETDO ;SET FDC-DONE BY ILLEGAL COMMAND

42 12715 006162 ERROR ;DONE ISN'T SET WITHIN 1 SEC.

43 12716 063477 SKPHN CPU

44 12717 000403 JMP .+3

45 12720 006162 ERROR ;FDC DIDN'T INTERRUPT AFTER SETTING DONE.

46 12721 000407 JMP 80010

47 12722 061477 INTA 0

48 12723 024213 LDA 1,F0077

49 12724 123400 AND 1,0

50 12725 024130 LDA 1,DEVICE

51 12726 106414 SUBO 0,1,SZR

52 12727 006162 ERROR ;ILLEGAL DEVICE CODE READ BY INTA AFTER

53 ;FDC-INTERRUPT. AC0= CODE READ, AC1=

54 ;EXPECTED CODE.

55 12730 006113 80010: LOOP

10064 .MAIN

```
01
02           ;TEST OF FDC MASK OUT LOGIC
03
04 12731 000402          JMP      B002
05
06 12732 177775 B002A: 177775
07
08 12733 000401 B002:   JMP      .+1
09 12734 006112          SETP2
10 12735 060277          INTDS
11 12736 020774          LDA      0,B002A
12 12737 062077          MSKO    0           ;MASK OUT FDC
13 12740 060177          INTEN
14 12741 006144          SETDO
15 12742 006162          ERROR
16 12743 065477          SKPBN   CPU
17 12744 006162          ERROR
18 12745 006113          LOOP
19
20
21           ;TEST OF INTERRUPT IDENT. REGISTER (SET AND CLEAR)
22           ;TEST OF STOP COMMAND
23
24 12746 000407          JMP      B003
25
26 12747 000000 B003A:  0
27 12750 012751 B003B:  .+1
28 12751 000010          000010
29 12752 000004          000004
30 12753 000002          000002
31 12754 000001          000001
32
33 12755 000401 B003:   JMP      .+1
34 12756 006112          SETP2
35 12757 020266          LDA      0,CURUN
36 12760 040767          STA      0,B003A ;SAVE CURRENT UNIT
37 12761 020267          LDA      0,MINDR
38 12762 040266 B0030:  STA      0,CURUN
39 12763 006144          SETDO
40 12764 006162          ERROR
41 12765 060461          DIA      0,XFDC ;EXEC ILLEGAL COMMAND
42 12766 034762          LDA      3,B003B ;DONE ISN'T SET AFTER ILL. COMMAND
43 12767 024266          LDA      1,CURUN ;GET INTERRUPT IDENT. REG.
44 12770 137000          ADD      1,3
45 12771 025400          LDA      1,0,3
46 12772 107415          AND0    0,1,SNK
47 12773 006162          ERROR
48
49
50 12774 130000          COM      1,2 ;INTERRUPT IDENT. REG. FAULTY. COR-
51 12775 113414          AND0    0,2,SRZ ;RECT DRIVE-BIT ISN'T SET. AC0= REG.
52 12776 006162          ERROR ;READ, AC1= EXPECTED REGISTER
53
54
55 12777 020266          LDA      0,CURUN ;OTHER INTERRUPT BIT THAN THE EXPEC-
56 13000 061061          DOA      0,XFDC ;TED ONE IS SET. AC0= IDENT.REG.
57 13001 063761          SKPDZ   XFDC ;READ, AC1= EXPECTED BIT.
58 13002 006162          ERROR ;SEND STOP COMMAND
59
59
59           ;FDC-DONE ISN'T CLEARED BY STOP-
59           ;COMMAND.
```

!0065 .MAIN

```
01
02 13005 060461      DIA      0,XFDC  ;GET INTR. IDENT. REG.
03 13006 101014      MOVØ     0,0,SZR
04 13005 006162      ERROR    ;INTERRUPT IDENT. REG. ISN'T CLEARED
05                                     ;BY STOP COMMAND
06 13006 020266      LDA      0,CURUN
07 13007 101400      INC      0,0
08 13010 024270      LDA      1,MAXDR
09 13011 106432      SUBZØ    0,1,SZC ;SKIP IF NEXT UNIT > 3
10 13012 000750      JMP      80030
11 13013 020734      LDA      0,8003A
12 13014 040266      STA      0,CURUN ;RESTORE CURRENT UNIT
13 13015 006113      LOOP
14
15 13016 102440      SUBØ     0,0
16 13017 006146      SETIN
17
18
19 13020 002177      JMP      &GRRET
20
21
22
23
24
25
26                                     ;FIRST TEST OF STATUS DELIVERED FROM FDC-CONTROLLER
27
28 13021 054177      GROUND: STA      3,GRRET
29
30 13022 000401      CØØØ:  JMP      .+1
31 13023 006111      SETP1
32 13024 102000      ADC      0,0
33 13025 034242      LDA      3,BASAD
34 13026 076061      DOB      3,XFDC ;SET BASE REGISTER
35 13027 024266      LDA      1,CURUN
36 13030 137000      ADD      1,3
37 13031 035400      LDA      3,0,3
38 13032 041405      STA      0,5,3 ;SET STATUS WORD = 177777
39 13033 006144      SETDØ    ;EXECUTE ILLEGAL COMMAND
40 13034 006162      ERROR    ;FDC-DONE ISN'T SET AFTER ILLEGAL
41                                     ;COMMAND.
42 13035 034242      LDA      3,BASAD
43 13036 024266      LDA      1,CURUN
44 13037 137000      ADD      1,3
45 13040 035400      LDA      3,0,3
46 13041 021405      LDA      0,5,3 ;ACØ:= STATUS RECEIVED
47
48 13042 126000      ADC      1,1
49 13043 106415      SUBØ     0,1,SNR
50 13044 006162      ERROR    ;NO STATUS IS DELIVERED VIA DATA
51                                     ;CHANNEL AFTER ILLEGAL COMMAND.
52 13045 024204      LDA      1,.186
53 13046 107415      ANDØ     0,1,SNR
54 13047 006162      ERROR    ;ILLEGAL COMMAND STATUS ISN'T SET
55                                     ;ACØ= RECEIVED STATUS
56 13050 006113      LOOP
57
58
59 13051 002177      JMP      &GRRET
```

10066 .MAIN

```
01
02          ;TEST OF CODE CONVERTER-CLEAR BY IO-RESET
03
04 13052 054177 GR003: STA      3,GRRET
05
06 13053 000402          JMP      0000
07
08 13054 000000 0000A: 0
09
10 13055 000401 0000:  JMP      .+1
11 13056 006110          SETPO
12 13057 062677          IORST          ;CLEAR CODE CONVERTER BY IO-RESET
13 13060 006147          INIACH          ;INITIALIZE ALL CHANNEL PROGRAMS
14 13061 006061          CWAIT          ;IORST WAIT
15 13062 000237          FD3KO
16
17 13063 054242          LDA      3,BASAD
18 13064 076061          DUB      3,XFDC ;SET BASE ADDRESS
19 13065 050266          LDA      2,CURUN
20 13066 157000          ADD      2,3
21 13067 035400          LDA      3,0,3
22 13070 020255          LDA      0,RCCOM ;GET READ CODE CONV. COMMAND
23 13071 041400          STA      0,0,3
24 13072 102000          ADC      0,0
25 13073 041405          STA      0,5,3 ;INITIALIZE STATUS WORD
26 13074 024210          LDA      1,.1B13 ;GET START BIT
27 13075 147000          ADD      2,1
28 13076 065061          DDA      1,XFDC ;START READ CODE CONV. COMMAND
29
30 13077 006145          WGSTATUS          ;WAIT FOR DONE AND GET STATUS TO ACO
31 13100 006162          ERROR          ;TIME OUT
32 13101 024264          LDA      1,STMSK
33 13102 107414          AND0    0,1,SZR
34 13103 006162          ERROR          ;STATUS ERROR, ACO= RECEIVED STATUS
35
36 13104 054262          LDA      3,RBFADR
37 13105 020266          LDA      0,CURUN
38 13106 117000          ADD      0,3
39 13107 035400          LDA      3,0,3
40 13110 050231          LDA      2,FD128
41 13111 150400          NEG      2,2
42 13112 126440          SUB0    1,1
43 13113 044741          STA      1,0000A
44
45 13114 121300 00000:  MOV0    1,0          ;CHECK CODE CONVERTER BY READING IT TO
46 13115 125400          INC      1,1          ;MEM.
47 13116 107000          ADD      0,1
48 13117 021400          LDA      0,0,3
49 13120 106415          SUB0    0,1,SNR
50 13121 000403          JMP      .+3
51 13122 006162          ERROR          ;CODE CONVERTER ISN'T CLEARED BY IO-RESET
52 13123 000412          JMP      00001
53 13124 175400          INC      3,3
54 13125 010727          ISZ     0000A
55 13126 010726          ISZ     0000A
56 13127 024725          LDA      1,0000A
57 13130 022341          LDA      0,RIEFLG
58 13131 101014          MOV0    0,0,SZR
59 13132 000403          JMP      00001
60 13133 151404          INC      2,2,SZR
61 13134 000760          JMP      00000
62 13135 006113 00001:  LOOP
```

10067 MAIN

```
01
02          ;TEST OF CODE CONVERTER LOAD AND READ
03
04 13136 000404      JMP      D001
05
06 13137 000000 D001A: 0
07 13140 000000 D001B: 0
08 13141 000377 D001C: 000377
09
10 13142 000401 D001:  JMP      +1
11 13143 006110      SETPO
12 13144 006147      INIACH          ;INITIALIZE CHANNEL PROGS.
13 13145 020774      LDA      0,D001C
14 13146 040771      STA      0,D001A
15
16 13147 034262 D0010: LDA      3,RBFADR
17 13150 024266      LDA      1,CURUN
18 13151 137000      ADD      1,3
19 13152 035400      LDA      3,0,3
20 13153 054257      STA      3,ARBUF
21 13154 034263      LDA      3,WBFADR
22 13155 137000      ADD      1,3
23 13156 035400      LDA      3,0,3
24 13157 054260      STA      3,AWBUF
25
26 13160 034260 D0011: LDA      3,AWBUF
27 13161 030231      LDA      2,FD128
28 13162 150400      NEG      2,2
29
30 13163 105300 D0012: MOVS     0,1      ;FILL WRITE BUFFER
31 13164 107000      ADD      0,1      ;FORM BOTH BYTES
32 13165 045400      STA      1,0,3    ;STORE 2 BYTES IN WRITE BUFFER
33 13166 175400      INC      3,3
34 13167 101400      INC      0,0
35 13170 151404      INC      2,2,SZR
36 13171 000772      JMP      D0012
37
38 13172 034242      LDA      3,BASAD
39 13173 076061      D0B      3,XFDC    ;SET BASE REG.
40 13174 024266      LDA      1,CURUN
41 13175 137000      ADD      1,3
42 13176 035400      LDA      3,0,3
43 13177 020260      LDA      0,AWBUF
44 13200 041404      STA      0,4,3    ;SET CH.PROG. BUFFER POINTER
45 13201 020250      LDA      0,LCCOM  ;GET LOAD CODE CONV. COMMAND
46 13202 041400      STA      0,0,3
47 13203 102000      ADC      0,0
48 13204 041405      STA      0,5,3    ;INITIALIZE STATUS WORD
49 13205 020210      LDA      0,.1B13
50 13206 107000      ADD      0,1
51 13207 065061      D0A      1,XFDC    ;START CH.PROG.
52
53 13210 006145      WGSTATUS          ;WAIT FOR DONE AND GET STATUS TO ACO
54 13211 006162      ERROR          ;TIME OUT
55 13212 024264      LDA      1,STMSK
56 13213 107414      AND0     0,1,SZR
57 13214 006162      ERROR          ;STATUS ERROR, ACO= RECEIVED STATUS
```

!0068 MAIN

```
01
02                                ;READ CODE CONVERTER BACK AGAIN
03 13215 020257 LDA 0,ARBUF
04 13216 024266 LDA 1,CURUN
05 13217 034242 LDA 3,BASAD
06 13220 076061 DOB 3,XFDC
07 13221 137000 ADD 1,3
08 13222 035400 LDA 3,0,3
09 13223 041404 STA 0,4,3 ;SET CH.PROG BUFFER POINTER
10 13224 020255 LDA 0,RCCOM
11 13225 041400 STA 0,0,3
12 13226 102000 ADC 0,0
13 13227 041405 STA 0,5,3 ;INITIALIZE STATUS WORD
14 13230 020210 LDA 0,.1B13
15 13231 107000 ADD 0,1
16 13232 065051 DOA 1,XFDC ;START CH.PROG.
17
18 13233 006145 WGSTAUS ;WAIT FOR DONE AND GET STATUS TO ACO
19 13234 006162 ERROR ;TIME OUT
20 13235 024264 LDA 1,STMSK
21 13236 107414 AND0 0,1,SZR
22 13237 006162 ERROR ;STATUS ERROR, ACO= RECEIVED STATUS
23
24 13240 020231 LDA 0,FD128
25 13241 040677 STA 0,DOU1B
26 13242 030260 LDA 2,AWBUF
27 13243 034257 LDA 3,ARBUF
28 13244 021400 D0013: LDA 0,0,3
29 13245 025000 LDA 1,0,2
30 13246 106415 SUB0 0,1,SNR
31 13247 000405 JMP .+5
32 13250 034260 LDA 3,AWBUF
33 13251 172400 SUB 3,2
34 13252 006162 ERROR ;ERROR IN CODE CONVERTER. CONTENT
35 ;READ <> DATA WRITTEN INTO CONV..ACO=
36 ;WORD READ (2 BYTES), AC1= WORD WRITTEN
37 ;AC2= WORD NO. WITH ERROR.
38 13253 000416 JMP D0014
39 13254 175400 INC 3,3
40 13255 151400 INC 2,2
41 13256 014662 DSZ D001B
42 13257 000765 JMP D0013
43
44 13260 022341 LDA 0,#IEFLG
45 13261 101014 MOV0 0,0,SZR
46 13262 000407 JMP D0014
47 13263 020654 LDA 0,DOU1A
48 13264 101400 INC 0,0
49 13265 040652 STA 0,DOU1A
50 13266 024214 LDA 1,F0377
51 13267 107414 AND0 0,1,SZR
52 13270 000670 JMP D0011
53 13271 006113 D0014: LOOP
```



```

0070 .MAIN
01
02 13355 034720 LDA 3,DC02C
03 13356 021402 LDA 0,2,3 ;GET BAD CYLINDERS READ.
04 13357 024714 LDA 1,0002A
05 13360 030714 LDA 2,0002B
06 13361 125300 MOVS 1,1
07 13362 147000 ADD 2,1
08 13363 106414 SUB# 0,1,SZR
09 13364 006162 ERROR ;THE BAD CYLINDERS READ <> BAD CYLINDERS
10 ;WRITTEN. AC0= BAD CYL.READ, AC1= BAD
11 ;CYL. WRITTEN.
12 13365 022341 LDA 0,#IEFLG
13 13366 101014 MOV# 0,0,SZR
14 13367 000412 JMP 00022
15 13370 020704 LDA 0,0002B
16 13371 024214 LDA 1,F0377
17 13372 106414 SUB# 0,1,SZR
18 13373 000403 JMP .+3
19 13374 020335 LDA 0,MXTRA
20 13375 000707 JMP 00021
21 13376 014676 DSZ 0002B
22 13377 020675 LDA 0,0002B
23 13400 000704 JMP 00021
24
25 13401 102000 00022: ADC 0,0 ;SET BAD CYL. = 377,377
26 13402 034673 LDA 3,0002C
27 13403 024247 LDA 1,LBCOM
28 13404 045400 STA 1,0,3 ;SET LOAD BAD CYL. COMMAND
29 13405 041401 STA 0,1,3
30 13406 024266 LDA 1,CURUN
31 13407 020210 LDA 0,.1B13
32 13410 107000 ADD 0,1
33 13411 065061 DOA 1,XFDC ;START CH. PROG.
34
35 13412 006145 WGSTATUS ;WAIT FOR DONE AND GET STATUS TO ACU
36 13413 006162 ERROR ;TIME OUT
37
38 13414 024264 LDA 1,STMSK
39 13415 107414 AND# 0,1,SZR
40 13416 006162 ERROR ;STATUS ERROR, AC0= STATUS RECEIVED
41
42 13417 006113 LOOP
43
44 13420 062677 IORST
45 13421 006061 CWAIT ;IORST WAIT
46 13422 000237 FDSK0
47
48
49 13423 002177 JMP AGRRET

```


!0072 .MAIN

```
01
02 13477 034242      LDA      3,BASAD
03 13500 137000      ADD      1,3
04 13501 035400      LDA      3,0,3
05 13502 021405      LDA      0,5,3 ;GET STATUS
06 13503 101014      MOVØ     0,0,ØZR
07 13504 006162      ERROR
08                                     ;STATUS ERROR. STATUS <> 0. ACO=
09                                     ;STATUS READ.
09 13505 006113      LOOP
10
11
12
13                                     ;FIRST TEST OF READ FUNCTION. SECTOR 0 (DOESN'T EXIST) IN TRACK
14                                     ;0 IS READ WITHOUT DATACHECK AT ALL.
15                                     ;FIRST TEST OF POSITION ERROR STATUS.
16
17 13506 000401      EQU2:   JMP      .+1
18 13507 006111      SETP1
19 13510 034242      LDA      3,BASAD
20 13511 076061      DOB      3,XFDC ;SET BASE REG.
21 13512 024266      LDA      1,CURUN
22 13513 137000      ADD      1,3
23 13514 035400      LDA      3,0,3
24 13515 020243      LDA      0,RDCOM
25 13516 041400      STA      0,0,3 ;SET READ COMMAND
26 13517 020224      LDA      0,FDO26
27 13520 101300      MOVS     0,0
28 13521 041401      STA      0,1,3 ;SET SECTORS PR. TRACK = 26
29                                     ;SET CYLINDER ADDRESS = 0
30 13522 102520      SUBZL   0,0 ;ACO:= 1
31 13523 041402      STA      0,2,3 ;SET SURFACE = 0
32                                     ;SET SECTOR ADDRESS = 0 (DOESN'T EXIST)
33                                     ;SET NO. OF SECTORS = 1
34 13524 020231      LDA      0,FD128
35 13525 041403      STA      0,3,3 ;SET NO OF BYTES = 128.
36 13526 030262      LDA      2,RBFADR
37 13527 031000      LDA      2,0,2
38 13530 051404      STA      2,4,3 ;SET DATA BUFFER ADDR.
39 13531 102440      SUBØ
40 13532 041405      STA      0,5,3 ;CLEAR STATUS WORD
41 13533 020210      LDA      0,.1B13
42 13534 107000      ADD      0,1
43 13535 065061      DOA      1,XFDC ;START READ COMMAND
44
45 13536 006063      TIMSK
46 13537 006654      3500.
47 13540 065661      SKPDN   XFDC
48 13541 006162      ERROR
49                                     ;FDC-DONE ISN'T SET AFTER RECAL.
50                                     ;COMMAND.
50 13542 024266      LDA      1,CURUN
51 13543 034242      LDA      3,BASAD
52 13544 137000      ADD      1,3
53 13545 035400      LDA      3,0,3
54 13546 021405      LDA      0,5,3 ;GET STATUS
55
56 13547 024207      LDA      1,.1B12
57 13550 107415      ANDØ    0,1,ØNR
58 13551 006162      ERROR
59                                     ;STATUS ERROR. POSITION BIT ISN'T SET
60                                     ;AFTER READ OF SECTOR 0, WHICH DOESN'T
61 13552 006113      LOOP
61                                     ;EXIST.
```

!0073 .MAIN

```
01
02 ;FIRST TEST OF READ OF AN EXISTING SECTOR. SECTOR 1 - 26 IN
03 ;TRACK 0 ARE READ WITHOUT ANY OTHER CHECK THAN STATUSCHECK.
04
05 13553 000402 JMP E009
06
07 13554 000000 E009A: 0 ;CURRENT SECTOR
08
09 13555 000401 E009: JMP .+1
10 13556 006110 SETPO
11 13557 102520 SUBZL 0,0
12 13560 040774 STA 0,E009A ;SET CURRENT SECTOR = 1
13 13561 006147 E0090: INIACH ;INITIALIZE CHANNEL PROGS.
14 13562 006161 SETBREG
15 13563 020243 LDA 0,RDCOM
16 13564 041400 STA 0,0,3 ;SET READ COMMAND
17 13565 020224 LOA 0,FD026
18 13566 101300 MOVS 0,0
19 13567 041401 STA 0,1,3 ;SET SECTORS PER TRACK= 26, CYL.ADDR.=
20 ;0.
21 13570 020764 LDA 0,E009A
22 13571 101300 MOVS 0,0
23 13572 152520 SUBZL 2,2
24 13573 143000 ADD 2,0
25 13574 041402 STA 0,2,3 ;SET SURFACE=0, SECTOR ADDR.= CURRENT,
26 ;NO. OF SECTORS = 1.
27 13575 020231 LDA 0,FD128
28 13576 041403 STA 0,3,3 ;SET NO. OF BYTES
29 13577 030262 LOA 2,RBFADR
30 13600 021000 LDA 0,0,2 ;USE READ BUFFER 0
31 13601 041404 STA 0,4,3 ;SET DATA BUFFER
32 13602 020210 LDA 0,.1H13
33 13603 107000 ADD 0,1
34 13604 065061 OUA 1,XFDC ;START CHANNEL PROG.
35
36 13605 006145 WGSTATUS ;WAIT FOR DONE AND GET STATUS TO AC0
37 13606 006162 ERROR ;TIME OUT
38 ;AC0= STATUS RECEIVED
39 13607 101015 MOV0 0,0,SNR
40 13610 000403 JMP .+3
41 13611 024743 LDA 1,E009A
42 13612 006162 ERROR ;RECEIVED STATUS <> 0. AC0= STATUS RE-
43 ;CEIVED, AC1= CURRENT SECTOR.
44 13613 022341 LDA 0,#IEFLG
45 13614 101014 MOV0 0,0,SZR
46 13615 000407 JMP E0091
47 13616 020736 LDA 0,E009A
48 13617 101400 INC 0,0 ;INCREMENT TO NEXT SECTOR
49 13620 040734 STA 0,E009A
50 13621 024224 LDA 1,FD026
51 13622 106432 SUBZ0 0,1,SZC ;SKIP IF NEW SECTOR > MAX SEC.
52 13623 000736 JMP E0090
53
54 13624 006113 E0091: LOOP
```

10074 .MAIN

```
01
02 ;FIRST TEST OF READ FROM CYLINDER <> U.
03 ;SECTOR 1-26 IN TRACK 1 ARE READ WITHOUT ANY OTHER CHECK THAN
04 ;STATUSCHECK.
05
06 13625 000402      JMP      E010
07
08 13626 000000 E010A: 0          ;CURRENT SECTOR
09
10 13627 000401 E010:  JMP      .+1
11 13630 006110      SETPO
12 13631 102520      SUBZL   0,0
13 13632 040774      STA      0,E010A ;SET CURRENT SECTOR = 1
14
15 13633 006147 E0100: INIACH          ;INITIALIZE CH. PROGS.
16 13634 006161      SETBREG
17
18 13635 020243      LDA      0,RDCOM
19 13636 041400      STA      0,0,3 ;SET READ COMMAND
20
21 13637 020334      LDA      0,MXSEC
22 13640 101300      MOVS    0,0
23 13641 152520      SUBZL   2,2
24 13642 143000      ADD     2,0
25 13643 041401      STA      0,1,3 ;SET SEC.PER TRACK, CYL.ADDR.=1
26
27 13644 020762      LDA      0,E010A
28 13645 101300      MOVS    0,0
29 13646 152520      SUBZL   2,2
30 13647 143000      ADD     2,0
31 13650 041402      STA      0,2,3 ;SET SURFACE=0,SEC.ADDR.=CURRENT, NO.
32 ;OF SECTORS= 1.
33 13651 020333      LDA      0,SECSIZ
34 13652 041403      STA      0,3,3 ;SET NO. OF BYTES
35
36 13653 030262      LDA      2,RBFADR
37 13654 021000      LDA      0,0,2 ;USE READ BUFFER 0
38 13655 041404      STA      0,4,3 ;SET DATA BUFFER ADDR.
39
40 13656 020210      LDA      0,.1B13
41 13657 107000      ADD     0,1
42 13660 065061      DOA     1,XFDC ;START CHANNEL PROG.
43
44 13661 006145      WGSTATUS
45 13662 006162      ERROR          ;TIME OUT
46
47 13663 101015      MOV0    0,0,SNR
48 13664 000403      JMP      .+3
49 13665 024741      LDA      1,E010A
50 13666 006162      ERROR          ;STATUS ERROR AFTER READ. AC0= STATUS,
51 ;AC1= SECTOR NO.
52 13667 022341      LDA      0,#IEFLG
53 13670 101014      MOV0    0,0,SZR
54 13671 000407      JMP      E0101
55 13672 020734      LDA      0,E010A
56 13673 101400      INC     0,0 ;INCREMENT TO NEXT SECTOR
57 13674 040732      STA      0,E010A
58 13675 024334      LDA      1,MXSEC
59 13676 106432      SUBZ0   0,1,SZC ;SKIP IF NEW SEC > MAX SEC.
60 13677 000734      JMP      E0100
61
62 13700 006113 E0101: LOOP
```

10075 .MAIN

```
01
02 ;TEST OF CORRECT TERMINATION VALUES OF: SECTOR ADDRESS, NO. OF
03 ;SECTORS, NO. OF BYTES, DATA BUFFER ADDRESS.
04
05 13701 000404 JMP E003
06
07 13702 002402 E003A: 002402
08 13703 000000 E003B: 0
09 13704 001000 E003C: 001000
10
11 13705 000401 E003: JMP .+1
12 13706 006111 SETP1
13 13707 006147 INIACH ;INITIALIZE CHANNEL PROGS.
14 13710 006161 SETBREG ;SET BASE REGISTER.
15 ;AC1= CUR.DRIVE, AC3= CH.PROG.ADDRESS
16 13711 020243 LDA 0,RDCOM
17 13712 041400 STA 0,0,3
18 13713 020334 LDA 0,MXSEC
19 13714 152520 SUBZL 2,2
20 13715 101300 MOVS 0,0
21 13716 143000 ADD 2,0
22 13717 040764 STA 0,E003B
23 13720 041401 STA 0,1,3 ;SET SECTORS PER TRACK = 26, CYLINDER-
24 ;ADDRESS = 1
25 13721 020761 LDA 0,E003A
26 13722 041402 STA 0,2,3 ;SET SURF.=0,SECTOR ADDR.= 5,NO.
27 ;OF SEC. = 2.
28 13723 020333 LDA 0,SECSIZ
29 13724 101120 MOVZL 0,0
30 13725 041403 STA 0,3,3 ;SET NO. OF BYTES = 256
31 13726 030262 LDA 2,RBFADR
32 13727 021000 LDA 0,0,2 ;USE READ BUFFER 0
33 13730 041404 STA 0,4,3
34 13731 020210 LDA 0,1B13
35 13732 107000 ADD 0,1
36 13733 065061 DOA 1,XFDC ;START CH. PROG
37
38 13734 006145 WGSTATUS ;WAIT FOR FDC-DONE AND GET STATUS TO ACO
39 13735 006162 ERROR ;TIME OUT
40
41 13736 101014 MOVW 0,0,SZR
42 13737 006162 ERROR ;STATUS <> 0, ACO= STATUS RECEIVED
43 13740 034242 LDA 3,BASAD
44 13741 024266 LDA 1,CURUN
45 13742 137000 ADD 1,3
46 13743 035400 LDA 3,0,3
47 13744 025401 LDA 1,1,3
48 13745 020736 LDA 0,E003B
49 13746 106414 SUBW 0,1,SZR
50 13747 006162 ERROR ;SEC. PER TRACK OR CYLINDER ADDRESS IS
51 ;CHANGED DURING TERM. OF CHA.PROG. ACO=
52 ;VALUES SET, AC1= VALUES AFTER TERM.
53 13750 020732 LDA 0,E003A
54 13751 024733 LDA 1,E003C
55 13752 123000 ADD 1,0 ;ADD NO. OF SECTORS TO SEC. ADDR.
56 13753 024216 LDA 1,F0002
57 13754 122400 SUB 1,0 ;SUBTRACT NO. OF SECTORS READ.
58 13755 025402 LDA 1,2,3
59 13756 106414 SUBW 0,1,SZR
60 13757 006162 ERROR ;ERROR IN TERM. VALUES OF SEC. ADDR. OR
61 ;NO. OF SECTORS. ACO= EXPECTED TERM. VA-
62 ;LUES, AC1= ACTUAL TERM. VALUES.
```

!0076 .MAIN

```
01
02 13760 021403      LDA      0,3,3
03 13761 101014      MOVØ     0,0,SZR
04 13762 006162      ERROR          ;BYTE COUNT <> 0 AFTER READ.
05 13763 030262      LDA      2,RBFADR
06 13764 021000      LDA      0,0,2
07 13765 024333      LDA      1,SECSIZ
08 13766 123000      ADD      1,0
09 13767 025404      LDA      1,4,3
10 13770 106414      SUBØ     0,1,SZR
11 13771 006162      ERROR          ;ERROR IN TERM. VALUE OF DATA BUFFER AD-
12                                     ;DRESS. ACO= EXPECTED ADDRESS, AC1= ACTU-
13                                     ;AL TERM. ADDRESS.
14 13772 006136      RECAL          ;RECALIBRATE DRIVE.
15
16 13773 006113      LOOP
17
18
19
20                                     ;FIRST TEST OF "BLOCK LENGTH ERROR" - STATUS.
21
22 13774 000403      JMP      E004
23
24 13775 000052 E004A: 42.
25 13776 000005 E004B: 5
26
27 13777 000401 E004:  JMP      .+1
28 14000 006111      SETP1
29
30 14001 006161      SETBREG
31 14002 020243      LDA      0,RDCOM
32 14003 041400      STA      0,0,3
33 14004 020771      LDA      0,E004A ;CYLINDER ADDRESS
34 14005 024771      LDA      1,E004B ;SECTOR ADDRESS
35 14006 123000      MOVS     1,1
36 14007 123000      ADD      1,0 ;ACO= SECTOR ADDR, CYLINDER ADDR.
37 14010 024217      LDA      1,F0003 ;AC1= 3 = NO OF SECTORS
38 14011 030333      LDA      2,SECSIZ ;AC2= 256 = NO OF BYTES
39 14012 151120      MOVZL   2,2
40
41 14013 006137      EXECUTE          ;SET CHANNEL PROG., START AND WAIT
42                                     ;FOR TERM.
43 14014 006162      ERROR          ;TIME OUT.
44                                     ;TERMINATED - ACO= STATUS
45
46 14015 024205      LDA      1,.1Bb
47 14016 107415      ANDØ     0,1,SNR
48 14017 006162      ERROR          ;BLOCK LENGTH ERROR - BIT ISN'T SET AF-
49                                     ;TER READ OF 3 SECTORS (EACH 128 BYTES)
50                                     ;WITH NO. OF BYTES SET TO 256.
51 14020 006113      LOOP
```

10077 .MAIN

```
01
02          ;TEST OF BASE ADDRESS REGISTER BY EXECUTING RECALIBRATE-COMMANDS
03          ;FROM DIFFERENT MEMORY AREAS.
04
05 14021 000404          JMP          E005
06
07 14022 000023AE005A:  HMEND          ;POINTER TO TOP OF MEMORY ADDRESS
08 14023 020000 E005B:  020000
09 14024 000000 E005C:  0          ;CURRENT BASE ADDRESS
10
11 14025 000401 E005:   JMP          .+1
12 14026 006111          SETP1
13 14027 020774          LDA          0,E005B
14 14030 040774          STA          0,E005C
15
16 14031 020773 E0051:  LDA          0,E005C
17 14032 024236          LDA          1,FD2KU
18 14033 123000          ADD          1,0          ;CALCULATE NEW CH. PROG. ADDRESS
19 14034 040770          STA          0,E005C
20 14035 024340          LDA          1,ATEND
21 14036 032764          LDA          2,E005A
22 14037 034231          LDA          3,FD128
23 14040 172400          SUB          3,2
24 14041 101120          MOVZL       0,0
25 14042 101220          MOVZR       0,0          ;REMOVE BIT 0
26 14043 112033          ADCZ0      0,2,SNC      ;SKIP IF ADDRESS < TOP OF MEMORY
27 14044 000455          JMP          E0052      ;EXIT
28 14045 106432          SUBZ0      0,1,SZC      ;SKIP IF ADDRESS > END OF PROGRAM
29 14046 000763          JMP          E0051
30 14047 115000          MOV          0,3
31 14050 076061          DOB          3,XFDC      ;SET BASE REG.
32 14051 024220          LDA          1,FD004
33 14052 123000          ADD          1,0
34 14053 041400          STA          0,0,3
35 14054 024222          LDA          1,FD006
36 14055 123000          ADD          1,0
37 14056 041401          STA          0,1,3
38 14057 123000          ADD          1,0
39 14060 041402          STA          0,2,3
40 14061 123000          ADD          1,0
41 14062 041403          STA          0,3,3
42 14063 024266          LDA          1,CURUN
43 14064 137000          ADD          1,3
44 14065 035400          LDA          3,0,3
45 14066 020246          LDA          0,RECOM
46 14067 041400          STA          0,0,3      ;SET RECAL-COMMAND
47 14070 102000          ADC          0,0
48 14071 041405          STA          0,5,3      ;INITIALIZE STATUS WORD
49 14072 020210          LDA          0,.1B13
50 14073 107000          ADD          0,1
51 14074 065061          DOA          1,XFDC      ;START CH. PROG
52
53 14075 006063          TIMSK          ;WAIT FOR DONE
54 14076 006654          3500.
55 14077 063661          SKPDN          XFDC
56 14100 006162          ERROR          ;TIME OUT
57
58 14101 006143          IDUNIT          ;IDENTIFY UNIT
59 14102 006162          ERROR          ;UNIT NOT IDENTIFIED.
60
61 14103 020266          LDA          0,CURUN
62 14104 106414          SUB0          0,1,SZR
63 14105 006162          ERROR          ;TERMINATION FROM WRONG UNIT.
```

```

0078 .MAIN
01
02 14106 034716      LDA      3,E005C
03 14107 171000      MOV      3,2
04 14110 117000      ADD      0,3
05 14111 035400      LDA      3,0,3
06 14112 021405      LDA      0,5,3 ;GET STATUS
07 14113 101014      MOVØ    0,0,SZR
08 14114 006162      ERROR
09
10
11
12
13
14 14115 022341      LDA      0,RIEFLG
15 14116 101014      MOVØ    0,0,SZR
16 14117 000402      JMP      E0052
17 14120 000711      JMP      E0051 ;CALCULATE NEXT BASE ADDRESS.
18
19 14121 020242 E0052: LDA      0,BASAD
20 14122 062061      DOB     0,XFDC ;INITIALIZE BASE ADDRESS REG.
21 14123 006113      LOOP
22
23
24
25 ;FIRST TEST OF THE COMMAND: READ ID-FIELDS
26
27 14124 000526      JMP      E0064 ;***** NOT WORKING IN RESENT CONTROLLERS
28 14125 000405      JMP      E006
29
30 14126 000000 E006A: 0 ;CURRENT CYLINDER
31 14127 000234 E006B: 156.
32 14130 000000 E006C: 0
33 14131 000000 E006D: 0
34
35 14132 000401 E006:  JMP      .+1
36 14133 024334      LDA      1,MXSEC
37 14134 030222      LDA      2,F0006
38 14135 006066      MULTI
39 14136 044771      STA      1,E006B ;AC1= MXSEC * 6
40 14137 006110      SETPU
41 14140 102520      SUBZL   0,0
42 14141 040765 E0061: STA      0,E006A ;CURRENT CYL. = 1
43
44 14142 006147      INIACH ;INITIALIZE CHANNEL PROGS.
45 14143 006161      SETBREG
46
47 14144 020253      LDA      0,RICOM
48 14145 041400      STA      0,0,3 ;SET READ ID-FIELDS COMMAND
49 14146 020334      LDA      0,MXSEC
50 14147 101300      MOVS    0,0
51 14150 030756      LDA      2,E006A
52 14151 143000      ADD      2,0
53 14152 041401      STA      0,1,3 ;SET SEC. PER TRACK, TRACK NO.
54 14153 020334      LDA      0,MXSEC
55 14154 041402      STA      0,2,3 ;SET NO. OF ID-FIELDS
56 14155 020752      LDA      0,E006B
57 14156 041403      STA      0,3,3 ;SET NO. OF BYTES
58 14157 030262      LDA      2,RBFADR
59 14160 021000      LDA      0,0,2
60 14161 041404      STA      0,4,3 ;SET READ BUFFER ADDR.
61 14162 020210      LDA      0,.1B13
62 14163 107000      ADD      0,1
63 14164 065061      DOA     1,XFDC ;START CHANNEL PROG.

```



```

U079 .MAIN
01
02 14165 006145      WGSTATUS      ;WAIT FOR DONE AND GET STATUS TO ACU
03 14166 006162      ERROR        ;TIME OUT
04
05 14167 101014      MOV@        0,0,SZR
06 14170 006162      ERROR        ;STATUS ERROR AFTER READ ID-FIELD COM-
07                                     ;MAND. ACU= RECEIVED STATUS.
08
09                                     ;CHECK THE 26 ID-FIELDS READ.
10 14171 030262      LDA        2,RRFADR
11 14172 031000      LDA        2,0,2
12 14173 020334      LDA        0,MXSEC
13 14174 040734      STA        0,E006C
14 14175 102520      SUBZL      0,0
15 14176 040733      STA        0,E006D
16
17 14177 021000      E0062: LDA    0,0,2 ;GET CYLINDER ADDRESS
18 14200 101300      MOVS      0,0
19 14201 034214      LDA        3,F0377
20 14202 163400      AND        3,0
21 14203 024723      LDA        1,E006A
22 14204 106414      SUB@      0,1,SZR
23 14205 006162      ERROR        ;ERROR IN ID-FIELD CYLINDER ADDR.
24                                     ;ACU= ADDRESS READ, AC1= EXPECTED
25                                     ;ADDRESS.
26 14206 021000      LDA        0,0,2 ;GET HEAD ADDRESS
27 14207 034214      LDA        3,F0377
28 14210 163400      AND        3,0
29 14211 101014      MOV@      0,0,SZR
30 14212 006162      ERROR        ;ERROR IN ID-FIELD HEAD ADDRESS.
31                                     ;ACU= HEAD ADDR. READ.
32 14213 021001      LDA        0,1,2 ;GET SECTOR ADDRESS
33 14214 101300      MOVS      0,0
34 14215 034214      LDA        3,F0377
35 14216 163400      AND        3,0
36 14217 024712      LDA        1,E006D
37 14220 106414      SUB@      0,1,SZR
38 14221 006162      ERROR        ;ERROR IN ID-FIELD SECTOR ADDRESS
39                                     ;ACU= SECTOR ADDR. READ, AC1= EX-
40                                     ;PECTED SECTOR ADDRESS.
41 14222 021001      LDA        0,1,2 ;GET SECTOR LENGTH
42 14223 034214      LDA        3,F0377
43 14224 163400      AND        3,0
44 14225 024333      LDA        1,SECSIZ
45 14226 125300      MOVS      1,1
46 14227 034217      LDA        3,FD003
47 14230 167400      AND        3,1
48 14231 106414      SUB@      0,1,SZR
49 14232 006162      ERROR        ;ERROR IN ID-FIELD SECTOR LENGTH.
50                                     ;ACU= SECTOR LENGTH READ. AC1= EXPEC-
51                                     ;TED LENGTH (0,1,2).
52 14233 020222      LDA        0,FD006
53 14234 113000      ADD        0,2
54 14235 022341      LDA        0,#IEFLG
55 14236 101014      MOV@      0,0,SZR
56 14237 000412      JMP        E0063
57 14240 010671      ISZ      E006D ;INCREMENT TO NEXT SECTOR
58 14241 014667      DSZ      E006C
59 14242 000735      JMP        E0062 ;CHECK NEXT ID-FIELD

```

10080 .MAIN

```
01
02 14243 020663      LDA      0,E006A ;ALL ID-FIELDS IN CUR. TRACK ARE CHECKED.
03 14244 101400      INC      0,0    ;INCREMENT TO NEXT TRACK
04 14245 024335      LDA      1,MAXTRA
05 14246 106432      SUBZ0   0,1,SZC ;SKIP IF NEW TRACK > MAX TRACK
06 14247 000672      JMP      E0061  ;CHECK NEXT TRACK
07 14250 006136      RECAL
08 14251 006113      E0063:  LOOP
09                      E0064:
```

!0081 .MAIN

01

02

;FIRST TEST OF REMAINING SECTOR - STATUS. TWO SECTORS HAS TO
;BE READ, STARTING WITH SECTOR 26.

03

04

05 14252 000401 E007: JMP .+1
06 14253 006111 SETP1
07 14254 006161 SETBREG
08 14255 020243 LDA 0,RDCOM
09 14256 041400 STA 0,0,3 ;SET READ COMMAND
10 14257 102520 SUBZL 0,0 ;ACU= 1 = CYL. ADDR.
11 14260 024334 LDA 1,MXSEC ;AC1= 26 = LAST SECTOR ON CYL.
12 14261 125300 MOVS 1,1
13 14262 123000 ADD 1,0 ;ACU= SECTOR ADDR, CYL. ADDR.
14 14263 024216 LDA 1,F0002 ;AC1= 2 = NO. OF SECTORS
15 14264 030333 LDA 2,SECSIZ ;AC2= 256 = NO. OF BYTES.
16 14265 151120 MOVZL 2,2

17

18 14266 006137 EXECUTE ;EXECUTE CH.PROG., WAIT FOR DONE

19 14267 006162 ERROR ;TIME OUT

20

;ACU= STATUS RECEIVED

21 14270 024201 LDA 1,.1B2

22 14271 107415 AND0 0,1,SNR

23 14272 006162 ERROR ;STATUS ERROR. REMAINING SECTORS
;SHOULD BE SET AFTER READING TWO SECTORS
;STARTING WITH SEC. ADDR. 26.

24

25

26 14273 006113 LOOP

27

28

29

;TEST OF BAD-CYLINDER ADMINISTRATION.

30

31 14274 000405 JMP E008

32

33 14275 000403 E008A: 000403

34 14276 002402 E008B: 002402

35 14277 002404 E008C: 002404

36 14300 000000 E008D: 0

37

38 14301 000401 E008: JMP .+1

39 14302 006111 SETP1

40 14303 020216 LDA 0,F0002

41 14304 040774 STA 0,E008D

42 14305 006161 SETBREG

43 14306 020247 LDA 0,LBCOM

44 14307 041400 STA 0,0,3 ;SET LOAD BAD CYL. COMMAND

45 14310 020765 LDA 0,E008A

46 14311 041401 STA 0,1,3 ;SET BAD CYLINDERS = 1 & 3

47 14312 020210 LDA 0,.1B13

48 14313 107000 ADD 0,1

49 14314 063061 DOA 1,XFDC ;START COMMAND

50

51 14315 006145 WGSTATUS ;WAIT FOR DONE AND GET STATUS TO ACU

52 14316 006162 ERROR ;TIME OUT

53

54 14317 024264 LDA 1,STMSK

55 14320 107414 AND0 0,1,SRZ

56 14321 006162 ERROR ;STATUS ERROR AFTER LOAD OF BAD CYL. COM
;ACU= STATUS RECEIVED

57

10082 .MAIN

```
01
02 14322 006161      SETBREG
03 14323 020245      LDA      0,VFCOM
04 14324 041400      STA      0,0,3      ;SET VERIFY COMMAND
05 14325 020751      LDA      0,E008B    ;AC0= SEC.ADDR (5), CYL. ADDR (2)
06
07 14326 126520 E0081: SUBZL    1,1      ;AC1= 1 = NO. OF SECTORS
08 14327 030353      LDA      2,SECSIZ    ;AC2= 128 = NO. OF BYTES
09
10 14330 006137      EXECUTE          ;EXECUTE CH.PROG AND WAIT FOR DONE
11 14331 006162      ERROR            ;TIME OUT
12                                     ;AC0= STATUS RECEIVED
13 14332 024207      LDA      1,.1B12
14 14333 107415      AND0     0,1,SNR
15 14334 006162      ERROR            ;STATUS ERROR. POSITION ERROR WASN'T SET
16                                     ;EVEN IF LOGICAL CYL. 2 WAS READ WITH
17                                     ;CYL.1 SET AS BAD CYLINDER.
18 14335 024202      LDA      1,.1B3
19 14336 107415      AND0     0,1,SNR
20 14337 006162      ERROR            ;STATUS ERROR. SEEK ERROR HAS TO BE SET
21                                     ;TOGETHER WITH BIT 12.
22 14340 020737      LDA      0,E008C
23 14341 014737      DSZ      E008D
24 14342 000764      JMP      E0081
25
26 14343 006113      LOOP
27 14344 006110      SETPO
28 14345 062677      IORST          ;CLEAR BAD. CYL. BY IO-RESET
29 14346 006061      CWAIT
30 14347 009237      FD3KO
31
32 14350 006161      SETBREG
33 14351 020245      LDA      0,VFCOM
34 14352 041400      STA      0,0,3      ;SET VERIFY COMMAND
35 14353 020724      LDA      0,E008C    ;AC0= SEC.ADDR.,CYL.ADDR.
36 14354 126520      SUBZL    1,1      ;AC1= 1 = NO. OF SECTORS
37 14355 030353      LDA      2,SECSIZ    ;AC2= 128 = NO. OF BYTES.
38
39 14356 006137      EXECUTE          ;EXECUTE CH.PROG., AND WAIT FOR DONE
40 14357 006162      ERROR            ;TIME OUT
41                                     ;AC0= STATUS RECEIVED
42 14360 024207      LDA      1,.1B12
43 14361 107414      AND0     0,1,SZR
44 14362 006162      ERROR            ;THE BAD CYLINDER ADMINISTRATION ISN'T
45                                     ;INITIALIZED BY IO-RESET.
46 14363 006136      RECAL
47 14364 006113      LOOP
48
49
50 14365 002177      JMP      &GRRET
```

!0083 .MAIN

```
01
02 14366 054177 GR005: STA      3,GRRET
03
04          ;READ OF SECTOR 7 & 8 IN ALL TRACKS. NO DATACHECK IS
05          ;PERFORMED. IF DOUBLE SIDE DISCETTES ARE USED, SECTOR 8 IS READ
06          ;FROM SURFACE 2.
07
08 14367 000407          JMP      F000
09
10 14370 000000 F000A:  0
11 14371 000000 F000B:  0          ;CURRENT TRACK
12 14372 000007 F000C:  7
13 14373 000010 F000D:  8.
14 14374 000000 F000E:  0          ;CURRENT SECTOR
15 14375 000200 F000F:  000200
16
17 14376 000401 F000:   JMP      .+1
18 14377 006110          SETPU
19 14400 006147          INIACH          ;INITIALIZE CHANNEL PROGS.
20 14401 126440          SUBO      1,1          ;AC1= 0 = FIRST TRACK
21
22 14402 044767 F0001:  STA      1,F000B
23 14403 006161          SETBREG
24 14404 020243          LDA      0,RDCOM
25 14405 041400          STA      0,0,3          ;SET READ COMMAND
26 14406 054762          STA      3,F000A          ;SAVE CH.PROG ADDR.
27 14407 030763          LDA      2,F000C
28
29 14410 050764 F0002:  STA      2,F000E          ;READ SECTOR
30 14411 020760          LDA      0,F000B
31 14412 024762          LDA      1,F000E
32 14413 125300          MOVS     1,1
33 14414 123000          ADD      1,0          ;AC0= SEC.ADDR., CYL.ADDR.
34 14415 126520          SURZL   1,1          ;AC1= 1 = NO. OF SECTORS
35 14416 030333          LDA      2,SECSIZ          ;AC2= 128 = NO. OF BYTES
36 14417 034752          LDA      3,F000B
37 14420 175015          MOVØ    3,3,SNR
38 14421 030231          LDA      2,FD128          ;IF TRACK 0 THEN SECSIZE= 128
39
40 14422 006137          EXECUTE          ;EXECUTE CH.PROG. AND WAIT FOR DONE.
41 14423 006162          ERROR          ;TIME OUT
42
43 14424 024264          LDA      1,STMSK
44 14425 107415          ANDØ    0,1,SNR
45 14426 000404          JMP      .+4
46 14427 024742          LDA      1,F000B
47 14430 030744          LDA      2,F000E
48 14431 006162          ERROR          ;STATUS ERROR AFTER READ. AC0= RECEI-
49          ;VED STATUS, AC1= TRACK NO., AC2= SEC-
50          ;TOR NO. (WITH SURFACE NO. = BIT 8)
51 14432 020742          LDA      0,F000E
52 14433 030740          LDA      2,F000D
53 14434 112033          ADCZØ   0,2,SNR          ;SKIP IF CUR.SECTOR < 8 (I.E. = 7)
54 14435 000407          JMP      F0003
55 14436 020271          LDA      0,MAXSU
56 14437 101015          MOVØ    0,0,SNR
57 14440 000750          JMP      F0002
58 14441 020754          LDA      0,F000F
59 14442 113000          ADD      0,2          ;ADD SURFACE BIT TO SECTOR NO.
60 14443 000745          JMP      F0002
```

```

10084 .MAIN
01
02 14444 032341 F0003: LDA      2,AIEFLG
03 14445 151014      MOVW    2,2,SZR
04 14446 000405      JMP      F0004
05 14447 030722      LDA      2,F000B
06 14450 006150      CALTR          ;CALCULATE NEW TRACK NO.
07 14451 000731      JMP      F0001      ;READ NEXT TRACK
08
09 14452 006136      RECAL          ;RECALIBRATE DRIVE
10 14453 006113 F0004: LOOP
11
12
13
14
15                ;FIRST TEST OF VERIFY SECTORS COMMAND.
16
17 14454 000403      JMP      F001
18
19 14455 000000 F001A: 0
20 14456 000000 F001B: 0
21
22 14457 000401 F001:  JMP      .+1
23 14460 006112      SETP2
24 14461 006161      SETBREG
25 14462 020245      LDA      0,VFCOM
26 14463 041400      STA      0,0,3      ;SET VERIFY COMMAND
27
28 14464 006135      RANTS          ;GET RANDOM TRACK- AND SECTOR NO.
29
30 14465 040770      STA      0,F001A
31 14466 044770      STA      1,F001B
32 14467 123000      ADD      1,0      ;AC0= SECTOR ADDR, CYL. ADDR.
33 14470 126520      SUBZL   1,1      ;AC1= 1 = NO. OF SECTORS
34 14471 030333      LDA      2,SECSIZ      ;AC2= 128 = NO. OF BYTES
35
36 14472 006137      EXECUTE          ;EXECUTE CH. PROG. AND WAIT FOR DONE
37 14473 006162      ERROR          ;TIME OUT
38                ;AC0= STATUS RECEIVED.
39 14474 101015      MOVW    0,0,SNR
40 14475 000405      JMP      .+5
41 14476 024757      LDA      1,F001A
42 14477 125300      MOVS    1,1
43 14500 030756      LDA      2,F001B
44 14501 006162      ERROR          ;STATUS ERROR AFTER VERIFY-COMMAND.
45                ;AC0= RECEIVED STATUS, AC1= SECTOR NO.
46                ;(WITH SURFACEBIT = BIT 8), AC2= TRACK
47                ;NO.
48 14502 006113      LOOP
49
50
51
52
53 14503 002177      JMP      &GRRET

```

10085 .MAIN

```
01
02 14504 054177 GR006: STA      3,GRRET
03
04          ;FIRST TEST OF READ FUNCTION WITH DATA-CHECK. THE LOADED DISCET-
05          ;TE(S) HAS TO BE THE FDC 705 TESTDISCETTE(S).
06          ;THIS DISCETTE DO ALWAYS HAVE 26 SECTORS PER TRACK.
07
08 14505 000406          JMP      G000
09
10 14506 000000 G000A: 0          ;CURRENT SECTOR (LEFT BYTE)
11 14507 000000 G000B: 0          ;CURRENT TRACK
12 14510 000074 G000C: 60.
13 14511 000000 G000D: 0
14 14512 052525 G000E: 052525    ;EXPECTED DATA-PATTERN.
15
16 14513 000401 G000G:  JMP      .+1
17 14514 006112          SETP2
18 14515 020206          LDA      0,CURUN
19 14516 006154          INISCH          ;INITIALIZE COMMUNICATION AREA.
20 14517 006161          SETBREG
21 14520 020243          LDA      0,RDCOM
22 14521 041400          STA      0,0,3    ;SET READ COMMAND
23 14522 006135          RANTS          ;GET RANDOM TRACK- AND SECTOR NO.
24 14523 040763          STA      0,G000A
25 14524 176520          SUBZL   3,3
26 14525 106400          SUB      3,1    ;TRACK:= TRACK-1
27 14526 044761          STA      1,G000B
28 14527 123000          ADD      1,0    ;ACU= SECTOR ADDR, CYL. ADDR
29 14530 126520          SUBZL   1,1    ;AC1= 1 = NO. OF SECTORS
30 14531 030231          LDA      2,FD128 ;AC2= 128 = NO. OF BYTES
31
32 14532 006137          EXECUTE          ;EXECUTE CH.PROG. AND WAIT FOR DONE
33 14533 006162          ERROR          ;TIME OUT
34
35 14534 101015          MOVW   0,0,SNR
36 14535 000405          JMP      .+5
37 14536 024750          LDA      1,G000A
38 14537 125300          MOVS   1,1
39 14540 030747          LDA      2,G000B
40 14541 006162          ERROR          ;STATUS ERROR AFTER READ. ACU= RECEIVED
41          ;STATUS,AC1= SECTOR NO., AC2= TRACK NO.
42 14542 030262          LDA      2,RBFADR
43 14543 024266          LDA      1,CURUN
44 14544 133000          ADD      1,2
45 14545 031000          LDA      2,0,2    ;AC2= ADDRESS OF DATABUFFER
46
47 14546 024266          LDA      1,CURUN
48 14547 021000          LDA      0,0,2    ;FIRST TWO BYTES READ
49 14550 106414          SUBW   0,1,SZR
50 14551 006162          ERROR          ;ERROR IN DATA READ. FIRST TWO BYTES
51          ;SHOULD BE UNIT NUMBER. ACU=
52          ;BYTES READ, AC1= EXPECTED BYTES.
53 14552 151400          INC      2,2
54 14553 024734          LDA      1,G000B
55 14554 021000          LDA      0,0,2
56 14555 106414          SUBW   0,1,SZR
57 14556 006162          ERROR          ;ERROR IN DATA READ. SECOND TWO BYTES
58          ;SHOULD BE CYLINDER NUMBER
59          ;ACU= BYTES READ, AC1= EXPECTED BYTES.
60 14557 151400          INC      2,2
```

!0086 .MAIN

```
01
02 14560 126440      SUBO    1,1
03 14561 034725      LDA     3,G000A
04 14562 175112      MOVLO  3,3,SZC ;SKIP IF HEAD 0
05 14563 126520      SUBZL  1,1    ;AC1= 1
06 14564 021000      LDA     0,0,2
07 14565 106414      SUBO   0,1,SZR
08 14566 006162      ERROR                      ;ERROR IN DATA READ. THIRD TWO BYTES
09                                     ;SHOULD BE HEAD NO. AC0= BYTES READ,
10                                     ;AC1= EXPECTED BYTES.
11 14567 151400      INC     2,2
12 14570 024716      LDA     1,G000A
13 14571 125120      MOVZL  1,1    ;REMOVE HEAD BIT
14 14572 125220      MOVZR  1,1
15 14573 125300      MOVS   1,1
16 14574 021000      LDA     0,0,2
17 14575 106414      SUBO   0,1,SZR
18 14576 006162      ERROR                      ;ERROR IN DATA READ. FOURTH TWO BYTES
19                                     ;SHOULD BE SECTOR NUMBER. AC0= BYTES
20                                     ;READ, AC1= EXPECTED BYTES.
21 14577 151400      INC     2,2
22 14600 020710      LDA     0,G000C
23 14601 040710      STA     0,G000D
24
25 14602 024710 G0001: LDA     1,G000E
26 14603 021000      LDA     0,0,2
27 14604 106415      SUBO   0,1,SNR
28 14605 000405      JMP     .+5
29 14606 030700      LDA     2,G000A
30 14607 034700      LDA     3,G000B
31 14610 173000      ADD     3,2
32 14611 006162      ERROR                      ;ERROR IN DATA READ. AC0= DATA READ (2
33                                     ;BYTES), AC1= EXPECTED DATA, AC2= SURFA-
34                                     ;CE, SECTOR ADDR., CYL. ADDR.
35 14612 036341      LDA     3,RIEFLG
36 14613 175014      MOVLO  3,3,SZR
37 14614 000404      JMP     G0002
38 14615 151400      INC     2,2
39 14616 014673      DSZ    G000D
40 14617 000763      JMP     G0001 ;CHECK NEXT BYTES.
41
42 14620 006113 G0002: LOOP
43
44
45 14621 002177      JMP     &GRRET
```


!0087 .MAIN

```
01
02 14622 054177 GR007: STA 3,GRRET
03
04 ;FIRST TEST OF WRITE FUNCTION. ALL SECTORS IN TRACK 0 ARE WRITTEN
05 ;WITH DUMMY PATTERNS. NO CHECK OF WRITTEN DATA IS PERFORMED.
06
07 14623 000402 JMP H000
08
09 14624 000000 H000A: 0 ;CURRENT SECTOR
10
11 14625 000401 H000: JMP +1
12 14626 006111 SETP1
13 14627 102520 SUBZL 0,0
14 14630 040774 STA 0,H000A
15
16 14631 020266 H0001: LDA 0,CURUN
17 14632 006154 INISCH ;INITIALIZE COMM. AREAS.
18 14633 006161 SETBREG ;SET UP BASE REG.
19
20 14634 020244 LDA 0,WRCOM
21 14635 041400 STA 0,0,3 ;SET WRITE COMMAND
22 14636 020224 LDA 0,FD026
23 14637 101300 MOVS 0,0
24 14640 041401 STA 0,1,3 ;SET SEC. PER TRACK, CYL. ADDR.
25 14641 030763 LDA 2,H000A
26 14642 151300 MOVS 2,2
27 14643 102520 SUBZL 0,0
28 14644 143000 ADD 2,0
29 14645 041402 STA 0,2,3 ;SET SURFACE, SECTOR ADDR., NO. OF SEC.
30 14646 020231 LDA 0,FD128
31 14647 041403 STA 0,3,3 ;SET NO. OF BYTES
32 14650 030263 LDA 2,WBFADR
33 14651 133000 ADD 1,2
34 14652 031000 LDA 2,0,2
35 14653 051404 STA 2,4,3 ;SET WRITE DATA BUFFER ADDRESS
36 14654 020210 LDA 0,1813
37 14655 107000 ADD 0,1
38 14656 065061 DOA 1,XFDC ;START CHANNEL PROG.
39
40 14657 006063 TIMSK ;WAIT FOR DONE (MAX 3.5 SEC)
41 14660 006054 3500.
42 14661 063661 SKPDN XFDC
43 14662 006162 ERROR ;TIME OUT. DONE ISN'T SET WITHIN 3.5
44 ;SEC. AFTER START OF A WRITE COMMAND.
45 14663 024266 LDA 1,CURUN
46 14664 034242 LDA 3,BASAD
47 14665 137000 ADD 1,3
48 14666 035400 LDA 3,0,3
49 14667 021405 LDA 0,5,3 ;GET STATUS
50 14670 101014 MOV0 0,0,SZR
51 14671 006162 ERROR ;STATUS ERROR AFTER WRITE. ACU= STATUS
52 ;RECEIVED.
53 14672 022341 LDA 0,AIEFLG
54 14673 101014 MOV0 0,0,SZR
55 14674 000407 JMP H0002
56 14675 020727 LDA 0,H000A
57 14676 101400 INC 0,0 ;INCREMENT TO NEXT SECTOR
58 14677 040725 STA 0,H000A
59 14700 024334 LDA 1,MXSEC
60 14701 106432 SUBZ0 0,1,SZC ;SKIP IF NEW SEC. > MXSEC
61 14702 000727 JMP H0001
62 14703 006113 H0002: LOOP
```

!0088 .MAIN

```
01
02 ;FIRST TEST OF WRITE-FUNCTION WITH FOLLOWING CHECK OF THE WRIT-
03 ;TEN DATA. A RANDOM SECTOR IS WRITTEN, THE HEAD IS MOVED, AND
04 ;THEN THE SECTOR IS CHECKREAD.
05
06 14704 000405      JMP      H001
07
08 14705 000000 H001A: 0          ;CURRENT SURFACE,SECTOR,CYLINDER
09 14706 000000 H001B: 0          ;CURRENT SECTOR
10 14707 000000 H001C: 0          ;CURRENT TRACK
11 14710 000000 H001D: 0
12
13 14711 000401 H001:  JMP      .+1
14 14712 006112      SETP2
15 14713 020266      LDA      0,CURUN
16 14714 006154      INISCH          ;INITIALIZE COMMUNICATION AREAS.
17
18 14715 030263      LDA      2,WBFADR
19 14716 113000      ADD      0,2
20 14717 031000      LDA      2,0,2
21 14720 050260      STA      2,AWBUF
22
23 14721 006142      PTFL      ;FILL WRITE-BUFFER WITH PATTERNS.
24
25 14722 006161      SETBREG
26 14723 020244      LDA      0,WPCOM
27 14724 041400      STA      0,0,3 ;SET WRITE COMMAND
28 14725 020263      LDA      0,WBFADR
29 14726 040261      STA      0,CURBF
30
31 14727 006135      RANTS          ;GET RANDOM SURF.,SECTOR AND TRACK NO.
32 14730 040756      STA      0,H001B
33 14731 044756      STA      1,H001C
34 14732 123000      ADD      1,0 ;ACU= SURFACE, SECTOR, TRACK
35 14733 040752      STA      0,H001A
36 14734 126520      SUBZL   1,1 ;AC1= 1 = NO. OF SECTORS
37 14735 030333      LDA      2,SECSIZ ;AC2= 128 = NO. OF BYTES
38
39 14736 006137      EXECUTE          ;EXECUTE CH.PROG AND WAIT FOR DONE
40 14737 006162      ERROR          ;TIME OUT
41
42 14740 101015      MOVW   0,0,SNR
43 14741 000405      JMP      .+5
44 14742 024744      LDA      1,H001B
45 14743 123300      MOVS   1,1
46 14744 030743      LDA      2,H001C
47 14745 006162      ERROR          ;STATUS ERROR AFTER WRITE. AC0= RECEI-
48 ;VED STATUS, AC1= SECTOR NO., AC2= TRACK
49 ;NO.
50 14746 006161      SETBREG
51 14747 020245      LDA      0,VFCOM
52 14750 041400      STA      0,0,3 ;SET VERIFY COMMAND
53 14751 020734      LDA      0,H001A
54 14752 024214      LDA      1,F0377
55 14753 107400      AND     0,1
56 14754 030333      LDA      2,MXTRA
57 14755 132432      SUBZ0   1,2,SZC ;SKIP IF NEW TRACK > 76
58 14756 101400      INC     0,0 ;INCREMENT TRACK NO. FOR HEAD MOVE
59 14757 126520      SUBZL   1,1
60 14760 006137      EXECUTE          ;EXECUTE CH.PROG AND WAIT FOR DONE
61 14761 006162      ERROR          ;TIME OUT
```

!0089 .MAIN

```
01
02 14762 006161      SETBREG
03 14763 020243      LDA      0,0,3      ;SET READ COMMAND
04 14764 041400      STA      0,0,3      ;SET READ COMMAND
05 14765 020720      LDA      0,H001A
06 14766 126520      SUBZL   1,1
07 14767 030333      LDA      2,SECSIZ
08
09 14770 006137      EXECUTE      ;EXECUTE CH.PROG AND WAIT FOR DONE
10 14771 006162      ERROR        ;TIME OUT
11
12 14772 101015      MOVØ     0,0,SNR
13 14773 000405      JMP      .+5
14 14774 024712      LDA      1,H001B
15 14775 125300      MOVS    1,1
16 14776 030711      LDA      2,H001C
17 14777 006162      ERROR        ;STATUS ERROR AFTER CHECKREAD. AC0= RE-
18                                     ;CEIVED STATUS,AC1= SECTOR NO.,AC2=
19                                     ;TRACK NO.
20 15000 022341      LDA      0,AIEFLG
21 15001 101014      MOVØ     0,0,SZR
22 15002 000426      JMP      H0012
23 15003 020333      LDA      0,SECSIZ
24 15004 101220      MOVZR   0,0
25 15005 040703      STA      0,H001D
26 15006 024260      LDA      1,CURUN
27 15007 034263      LDA      3,WBFADR
28 15010 030262      LDA      2,WBFADR
29 15011 137000      ADD     1,3
30 15012 133000      ADD     1,2
31 15013 035400      LDA      3,0,3
32 15014 031000      LDA      2,0,2
33                                     ;COMPARE DATA WRITTEN WITH DATA READ.
34 15015 021400 H0011: LDA      0,0,3
35 15016 025000      LDA      1,0,2
36 15017 106415      SUBØ    0,1,SNR
37 15020 000403      JMP      .+3
38 15021 006162      ERROR        ;DATA WRITTEN <> DATA READ. AC0= DATA
39                                     ;WRITTEN (2 BYTES), AC1= DATA READ.
40 15022 000406      JMP      H0012
41 15023 173400      INC     3,3
42 15024 151400      INC     2,2
43 15025 014603      DSZ    H001D      ;SKIP IF ALL BYTES ARE COMPARED.
44 15026 000767      JMP      H0011      ;COMPARE NEXT BYTES
45
46 15027 006151      COMPT      ;COMPLEMENT DATAPATTERN IN WRITE-BUFFER.
47 15030 006113 H0012: LOOP
```

10090 .MAIN

```
01
02          ;CHECK OF HEAD SELECTION.
03          ;ONLY EXECUTED WHEN DOUBLE-SIDED DISCETTES ARE USED.
04
05 15031 000404          JMP      H002
06
07 15032 002425 H002A: 002425          ;SIDE 0, SECTOR 5, TRACK 21
08 15033 102425 H002B: 102425          ;SIDE 1, SECTOR 5, TRACK 21
09 15034 000000 H002C: 0              ;CURRENT SIDE
10
11 15035 000401 H002:  JMP      .+1
12 15036 020271          LDA      0,MAXSU
13 15037 101015          MOVW   0,0,SNR
14 15040 000513          JMP      H0029 ;EXIT IF SINGLE SIDE
15
16 15041 006111          SETP1
17 15042 102440          SUBW   0,0
18 15043 040771          STA      0,H002C ;CLEAR CURRENT SURFACE
19 15044 020266          LDA      0,CURUN
20 15045 006154          INISCH ;INITIALIZE CH. PROGS.
21
22 15046 030263 H0020:  LDA      2,WBFADR
23 15047 024266          LDA      1,CURUN
24 15050 133000          ADD      1,2
25 15051 031000          LDA      2,0,2
26 15052 020762          LDA      0,H002C
27 15053 126440          SUBW   1,1
28 15054 101014          MOVW   0,0,SZR
29 15055 126000          ADC      1,1
30 15056 045000          STA      1,0,2 ;SET FIRST DATA WORD = 0 OR -1
31
32 15057 006161          SETBREG
33
34 15060 020244          LDA      0,WRCOM
35 15061 041400          STA      0,0,3 ;SET WRITE COMMAND
36 15062 020263          LDA      0,WBFADR
37 15063 040261          STA      0,CURBF
38 15064 020746          LDA      0,H002A
39 15065 024747          LDA      1,H002C
40 15066 125014          MOVW   1,1,SZR
41 15067 020744          LDA      0,H002B
42 15070 126520          SUBZL  1,1
43 15071 030333          LDA      2,SECSIZ
44
45 15072 006137          EXECUTE ;EXECUTE WRITE COMMAND
46 15073 006162          ERROR  ;TIME OUT
47
48 15074 101014          MOVW   0,0,SZR
49 15075 006162          ERROR  ;STATUS ERROR AFTER WRITE. ACO= STATUS
50                                     ;RECEIVED.
51
52 15076 020736          LDA      0,H002C
53 15077 101014          MOVW   0,0,SZR
54 15100 000404          JMP      H0021 ;SURFACE 0 AND 1 IS WRITTEN
55 15101 102520          SUBZL  0,0
56 15102 040732          STA      0,H002C ;SET CURRENT SURFACE = 1
57 15103 000743          JMP      H0020 ;WRITE SURFACE 1
```

!0091 .MAIN:

```
01
02 15104 102440 H0021: SUB0 0,0
03 15105 040727 STA 0,H002C ;CLEAR CURRENT SURFACE
04
05 15106 030262 H0022: LDA 2,RBFADR
06 15107 050261 STA 2,CURBF
07
08 15110 006161 SETBREG
09
10 15111 020243 LDA 0,RDCOM
11 15112 041400 STA 0,0,3 ;SET READ COMMAND
12 15113 020717 LDA 0,H002A
13 15114 024720 LDA 1,H002C
14 15115 125014 MOV0 1,1,SZR
15 15116 020715 LDA 0,H002B
16 15117 126520 SUBZL 1,1
17 15120 030333 LDA 2,SECSIZ
18
19 15121 006137 EXECUTE ;EXECUTE READ COMMAND
20 15122 006162 ERROR ;TIME OUT
21
22 15123 101014 MOV0 0,0,SZR
23 15124 006162 ERROR ;STATUS ERROR AFTER READ. ACQ= STATUS
24 ;RECEIVED.
25 15125 030262 LDA 2,RBFADR
26 15126 024266 LDA 1,CURUN
27 15127 133000 ADD 1,2
28 15130 031000 LDA 2,0,2
29 15131 021000 LDA 0,0,2 ;ACQ= FIRST DATA WORD READ
30
31 15132 024702 LDA 1,H002C
32 15133 125014 MOV0 1,1,SZR
33 15134 000405 JMP .+5
34 15135 126000 ADC 1,1
35 15136 106415 SUB0 0,1,SNR
36 15137 006162 ERROR ;SIDE 1 IS READ EVEN IF THE COMMAND WAS
37 ;READ SIDE 0.
38 15140 000404 JMP .+4
39 15141 126440 SUB0 1,1
40 15142 106415 SUB0 0,1,SNR
41 15143 006162 ERROR ;SIDE 0 IS READ EVEN IF THE COMMAND WAS
42 ;READ SIDE 1.
43 15144 024670 LDA 1,H002C
44 15145 125014 MOV0 1,1,SZR
45 15146 000404 JMP H0023 ;SURFACE 0 & 1 ARE READ.
46
47 15147 102520 SUBZL 0,0
48 15150 040664 STA 0,H002C
49 15151 000735 JMP H0022 ;READ SURFACE 1
50
51 15152 006113 H0023: LOOP
52 H0029:
53
54
55 15153 002177 JMP #GRRET
```

10092 .MAIN

```
01
02 15154 054177 GR010: STA      3,GRRET
03
04          ;LOOP TO INITIALIZE THE DISCETTE TO IBM-FORMAT.
05
06 15155 000420          JMP      K000
07
08 15156 000374 K000A: 000374          ;HEX = FC
09 15157 000376 K000B: 000376          ;HEX = FE
10 15160 000367 K000C: 000367          ;HEX = F7
11 15161 000373 K000D: 000373          ;HEX = FB
12 15162 000345 K000G: 000345
13 15163 015417 K000H: K0009
14 15164 000000 K000E: 0              ;CURRENT TRACK
15 15165 000000 K000F: 0              ;CURRENT SURFACE
16 15166 000000 K000I: 0              ;CURRENT SECTOR
17 15167 000200 K000J: 128.          ;SECTOR LENGTH - SET DURING INIT.
18 15170 000052 K000K: 26.           ;LAST SECTOR - SET DURING INIT.
19 15171 015172 K000L: .+1          ;TABLE WITH LENGTH'S OF GAPS
20 15172 000041          33.
21 15173 000060          48.
22 15174 000077          63.
23
24 15175 000401 K000:   JMP      .+1
25 15176 006110          SETPU
26 15177 102440 K000X:  SUBO      0,0
27 15200 040765          STA      0,K000F ;CLEAR CURRENT SURFACE
28 15201 102440 K000Y:  SUBO      0,0
29 15202 040762          STA      0,K000E ;CLEAR CURRENT TRACK
30 15203 102520 K000Z:  SUBZL     0,0
31 15204 040762          STA      0,K000I ;SET FIRST SECTOR
32
33          ;PLACE CORE IMAGE OF TRACK START. "TSEND"
34 15205 030340          LDA      2,ATEND
35 15206 102520          SUBZL     0,0
36 15207 112400          SUB      0,2      ;AC2= TSEND-1
37 15210 151120          MOVZL    2,2      ;AC2= BYTE COUNTER
38
39          ;START OF GAP 5
40 15211 024226          LDA      1,F0040
41 15212 124400          NEG      1,1
42 15213 020214          LDA      0,F0377
43 15214 006747          JSR      4,K000H ;PUT 40 * FF
44 15215 125404          INC      1,1,SZR
45 15216 000775          JMP      .-3
46
47 15217 024222          LDA      1,F0006
48 15220 124400          NEG      1,1
49 15221 102440          SUBO      0,0
50 15222 006741          JSR      4,K000H ;PUT 6 * 00
51 15223 125404          INC      1,1,SZR
52 15224 000775          JMP      .-3
53
54          ;SET INDEX ADDRESS MARK
55 15225 020731          LDA      0,K000A
56 15226 004571          JSR      K0009 ;PUT BYTE (FC)
```

!0093 .MAIN

```
01
02                                ;START OF GAP 1
03 15227 024224 LDA 1,F0026
04 15230 124400 NEG 1,1
05
06 15231 020214 LDA 0,F0377
07 15232 004565 JSR K0009 ;PUT 26 * FF
08 15233 125404 INC 1,1,SZR
09 15234 000775 JMP .-3
10
11 15235 024222 LDA 1,F0006
12 15236 124400 NEG 1,1
13
14 15237 102440 SUBO 0,0
15 15240 004557 JSR K0009 ;PUT 6 * 00
16 15241 125404 INC 1,1,SZR
17 15242 000775 JMP .-3
18
19                                ;START OF ID-FIELD
20 15243 020714 K0000: LDA 0,K000B
21 15244 004553 JSR K0009 ;PUT BYTE (FE)
22
23 15245 020717 LDA 0,K000E
24 15246 004551 JSR K0009 ;PUT BYTE (CYLINDER ADDRESS)
25
26 15247 020716 LDA 0,K000F
27 15250 004547 JSR K0009 ;PUT BYTE (HEAD ADDRESS)
28
29 15251 020715 LDA 0,K000I
30 15252 004545 JSR K0009 ;PUT BYTE (SECTOR ADDRESS)
31
32 15253 020714 LDA 0,K000J
33 15254 034710 LDA 3,K000E
34 15255 175015 MOVW 3,3,SNR
35 15256 020231 LDA 0,F0128 ;IF TRACK 0 THEN SEC.SIZE = 128
36 15257 101300 MOVS 0,0
37 15260 034217 LDA 3,F0003
38 15261 163400 AND 3,0 ;ACU= SECTOR LENGTH IDENT. (0,1,2)
39 15262 004535 JSR K0009 ;PUT BYTE (SECTOR LENGTH)
40
41 15263 020675 LDA 0,K000C
42 15264 004533 JSR K0009 ;PUT BYTE (F7)
43
44                                ;START OF GAP 2
45 15265 024223 LDA 1,F0011
46 15266 124400 NEG 1,1
47
48 15267 020214 LDA 0,F0377
49 15270 004527 JSR K0009 ;PUT 11 * FF
50 15271 125404 INC 1,1,SZR
51 15272 000775 JMP .-3
52
53 15273 024222 LDA 1,F0006
54 15274 124400 NEG 1,1
55
56 15275 102440 SUBO 0,0
57 15276 004521 JSR K0009 ;PUT 6 * 00
58 15277 125404 INC 1,1,SZR
59 15300 000775 JMP .-3
```

10094 .MAIN

```
01
02                                ;START OF DATA FIELD
03 15301 020660      LDA      0,K000D
04 15302 004515      JSR      K0009      ;PUT BYTE (F6)
05
06 15303 024664      LDA      1,K000J
07 15304 034660      LDA      3,K000E
08 15305 175015      MOVW    3,3,SNR
09 15306 024231      LDA      1,FD128 ;IF TRACK 0 THEN SEC.SIZE = 128
10 15307 124400      NEG      1,1
11
12 15310 020652      LDA      0,K000G
13 15311 004506      JSR      K0009      ;PUT 128 * DATA (OCTAL - 345)
14 15312 125404      INC      1,1,SZR
15 15313 000775      JMP      .-3
16
17 15314 020644      LDA      0,K000C
18 15315 004502      JSR      K0009      ;PUT BYTE (F7)
19
20                                ;START OF GAP 3
21 15316 102440      SUBW   0,0
22 15317 024650      LDA      1,K000J ;AC1= SECTOR LENGTH
23 15320 034231      LDA      3,FD128
24 15321 136415      SUBW   1,3,SNR
25 15322 000405      JMP      .+5      ;128
26 15323 101400      INC      0,0
27 15324 034232      LDA      3,FD256
28 15325 136414      SUBW   1,3,SZR
29 15326 101400      INC      0,0      ;512
30                                ;256
31
32 15327 034642      LDA      3,K000L
33 15330 117000      ADD     0,3
34 15331 025400      LDA      1,0,3      ;AC1= CORRESPONDING GAP-LENGTH
35 15332 034632      LDA      3,K000E
36 15333 175015      MOVW    3,3,SNR
37 15334 024636      LDA      1,K000L+1 ;IF TRACK 0 THEN GAPLENGT = 33
38 15335 020222      LDA      0,FD006
39 15336 106400      SUB     0,1
40 15337 124400      NEG      1,1
41
42 15340 020214      LDA      0,FD0377
43 15341 004456      JSR      K0009      ;PUT N3 * FF
44 15342 125404      INC      1,1,SZR
45 15343 000775      JMP      .-3
46
47 15344 024222      LDA      1,FD006
48 15345 124400      NEG      1,1
49
50 15346 102440      SUBW   0,0
51 15347 004450      JSR      K0009      ;PUT 6 * 00
52 15350 125404      INC      1,1,SZR
53 15351 000775      JMP      .-3
```


!0095 .MAIN

```
01
02 15352 020614 LDA 0,K0001 ;CURRENT SECTOR
03 15353 101400 INC 0,0 ;INCREMENT TO NEXT SECTOR
04 15354 040612 STA 0,K0001
05 15355 024613 LDA 1,K000K
06 15356 036462 LDA 3,K0000
07 15357 175015 MOV0 3,3,SNR
08 15360 024224 LDA 1,F0026 ;IF TRACK 0 THEN 0 SECTORS = 26
09 15361 106432 SUBZ0 0,1,SZC ;SKIP IF NEW SEC. > LAST SECTOR.
10 15362 000661 JMP K0000 ;WRITE NEXT SECTOR
11
12 ;START OF GAP 4
13 15363 024235 LDA 1,F01K0
14 15364 124400 NEG 1,1
15
16 15365 020214 LDA 0,F0377
17 15366 004431 JSR K0009 ;PUT 400 * FF
18 15367 125404 INC 1,1,SZR
19 15370 000775 JMP .-3
20
21 ;GEN. OF TRACK-IMAGE FINISHED
22 15371 004453 JSR K0008 ;CALL WRITE TRACK PROCEDURE.
23
24 15372 022446 K0001: LDA 0,K0000 ;CURRENT TRACK
25 15373 101400 INC 0,0 ;INCREMENT TO NEXT TRACK
26 15374 042444 STA 0,K0000
27 15375 024335 LDA 1,MXTRA
28 15376 106432 SUBZ0 0,1,SZC ;SKIP IF NEW TRACK > MAX. TRACK
29 15377 002443 JMP AK000G ;WRITE NEXT TRACK
30
31 15400 020271 LDA 0,MAXSU
32 15401 101015 MOV0 0,0,SNR
33 15402 000412 JMP K0005 ;SINGLE SIDE DISCETTE
34 15403 022436 LDA 0,K000P ;GET CURRENT SURFACE
35 15404 101014 MOV0 0,0,SZR
36 15405 000407 JMP K0003 ;SURFACE 1 IS WRITTEN
37 15406 102520 SUBZL 0,0
38 15407 042432 STA 0,K000P ;SET CURRENT SURFACE = 1
39 15410 006136 RECAL
40 15411 006061 CWAIT
41 15412 000233 F0350
42 15413 002430 JMP AK000R ;WRITE SURFACE 1
43
44 15414 006136 K0003: RECAL
45 15415 006113 LOOP
46 15416 000512 JMP K0002
```

!0096 .MAIN

```
01
02 15417 054417 K0009: STA 3,K000M ;PROCEDURE TO PUT BYTES
03 15420 034214 LDA 3,F0377
04 15421 163400 AND 3,0
05 15422 151212 MOVW 2,2,SZC ;TEST BYTE POINTER
06 15423 000405 JMP .+5
07
08 15424 101300 MOVS 0,0 ;SWAP BYTE
09 15425 040412 STA 0,K000N ;SAVE UNTIL NEXT BYTE
10 15426 151400 INC 2,2 ;INCREMENT BYTE POINTER
11 15427 002407 JMP K000M
12
13 15430 034407 LDA 3,K000N ;STORE 2 BYTES
14 15431 163000 ADD 3,0
15 15432 151400 INC 2,2
16 15433 155220 MOVZR 2,3 ;FORM WORD ADDRESS
17 15434 041400 STA 0,0,3
18 15435 002401 JMP K000M
19
20 15436 000000 K000M: 0
21 15437 000000 K000N: 0
22 15440 015164 K0000: K000E
23 15441 015165 K000P: K000F
24 15442 015203 K000Q: K000Z
25 15443 015201 K000R: K000Y
26
27 15444 054772 K0008: STA 3,K000M ;PROCEDURE TO SET TRACK FORMAT
28 15445 034242 LDA 3,BASAD ;CHANNEL COMMAND AND EXECUTE IT
29 15446 076061 DOB 3,XFDC
30 15447 024266 LDA 1,CURM
31 15450 137000 ADD 1,3
32 15451 035400 LDA 3,0,3
33 15452 020252 LDA 0,TFCOM
34 15453 041400 STA 0,0,3 ;SET TRACK FORMAT COMMAND
35 15454 022764 LDA 0,K0000
36 15455 041401 STA 0,1,3 ;SET CYLINDER ADDRESS
37 15456 102440 SUBO 0,0
38 15457 032762 LDA 2,K000P
39 15460 151014 MOVW 2,2,SZR ;SKIP IF SURFACE 0
40 15461 020200 LDA 0,.1B0
41 15462 041402 STA 0,2,3 ;SET SURFACE
42 15463 020240 LDA 0,F05K5
43 15464 041403 STA 0,3,3 ;SET NO. OF BYTES
44 15465 020340 LDA 0,ATEND
45 15466 041404 STA 0,4,3 ;SET DATA BUFFER ADDRESS
46 15467 102000 ADC 0,0
47 15470 041405 STA 0,5,3 ;INITIALIZE STATUS WORD
48
49 15471 020210 LDA 0,.1B13
50 15472 107000 ADD 0,1
51 15473 065061 DOA 1,XFDC ;START CHANNEL PROGRAM
52
53 15474 006145 WGSTATUS ;WAIT FOR DONE AND GET STATUS TO ACO
54 15475 006162 ERROR ;TIME OUT
```

10097 .MAIN

```
01
02 15476 034203 LDA 3,1B5
03 15477 117414 AND0 0,3,SZR ;SKIP IF NOT WRITE PROTECT
04 15500 000415 JMP K0004
05 15501 101014 MOV0 0,0,SZR
06 15502 006162 ERROR ;STATUS ERROR. ACO= RECEIVED STATUS
07 15503 022735 LDA 0,K0000 ;GET CURRENT TRACK
08 15504 101014 MOV0 0,0,SZR
09 15505 000404 JMP .+4
10 15506 006136 RECAL
11 15507 006061 CWAIT
12 15510 000233 FD350
13 15511 022341 LDA 0,AIEFLG
14 15512 101014 MOV0 0,0,SZR
15 15513 000701 JMP K0005
16 15514 002722 JMP K0000M
17
18
19 15515 060277 K0004: INTDS
20 15516 006043 CCRLF
21 15517 006046 CDICL
22 15520 006040 CMES
23 15521 010644 DWPRO
24 15522 006044 CDISP
25 15523 010644 DWPRO
26 15524 006050 CHAAT
27 15525 006006 RESET
28 15526 006136 RECAL
29 15527 000531 JMP K0029
30
31
32 K0002:
```

!0098 .MAIN

```
01
02          ;VERIFY ALL SECTORS AFTER DISCETTE FORMATING.
03
04 15530 000410          JMP      K001
05
06 15531 000000 K001A: 0          ;CURRENT SURFACE
07 15532 000000 K001B: 0          ;CURRENT CYLINDER
08 15533 000001 K001C: 000001
09 15534 000400 K001D: 000400
10 15535 100400 K001E: 100400
11 15536 015170 K001F: K000K
12 15537 000000 K001G: 0
13
14 15540 000401 K001:  JMP      .+1
15 15541 006110          SETPO
16 15542 102440          SUBO    0,0
17 15543 040766          STA    0,K001A ;CLEAR CURRENT SURFACE
18 15544 040766          STA    0,K001B ;CLEAR CURRENT CYLINDER
19
20 15545 006161 K0011: SETBREG
21 15546 020245          LDA    0,VFCOM
22 15547 041400          STA    0,0,3 ;SET VERIFY COMMAND
23 15550 020764          LDA    0,K001D
24 15551 034760          LDA    3,K001A
25 15552 175014          MOV#   3,3,SZR ;SKIP IF SURFACE 0
26 15553 020762          LDA    0,K001E
27 15554 024756          LDA    1,K001B
28 15555 123000          ADD   1,0 ;AC0= SURFACE,SECTOR ADDR.,CYL.ADDR.
29 15556 026760          LDA    1,K001F ;AC1= 26 = NO OF SECTORS
30 15557 034753          LDA    3,K001B
31 15560 175015          MOV#   3,3,SNR
32 15561 024224          LDA    1,FDU26 ;IF TRACK 0 THEN # SECTORS = 26
33                                     ;AC2= DUMMY IN VERIFY-COMMANDS.
34
35 15562 034334          LDA    3,MXSEC
36 15563 054754          STA    3,K001G
37 15564 044334          STA    1,MXSEC
38 15565 006137          EXECUTE ;EXECUTE CH.PROG. AND WAIT FOR DONE
39 15566 006162          ERROR   ;TIME OUT
40                                     ;AC0= STATUS RECEIVED
41 15567 034750          LDA    3,K001G
42 15570 054334          STA    3,MXSEC
43
44 15571 101015          MOV#   0,0,SNR
45 15572 000405          JMP      .+5
46 15573 024737          LDA    1,K001B
47 15574 030735          LDA    2,K001A
48 15575 006162          ERROR   ;VERIFY ERROR AFTER DISCETTE FORMATING.
49                                     ;AC0= STATUS READ, AC1= CYLINDER ADDR.,
50                                     ;AC2= SURFACE.
51 15576 000422          JMP      K0013
52 15577 020271          LDA    0,MAXSU
53 15600 101015          MOV#   0,0,SNR ;IF MORE THAN 1 SURFACE THEN INCREMENT TO
54 15601 000407          JMP      K0012 ;NEXT.
55 15602 020727          LDA    0,K001A
56 15603 101014          MOV#   0,0,SZR
57 15604 000404          JMP      K0012
58 15605 102520          SUBZL  0,0 ;AC0= 1
59 15606 040723          STA    0,K001A ;SET CURRENT SURFACE = 1
60 15607 000736          JMP      K0011
```

!0099 .MAIN

```
01
02 15610 102440 KU012: SUBO    0,0
03 15611 040720      STA    0,K001A
04 15612 020720      LDA    0,K001B ;INCREMENT TO NEXT CYLINDER
05 15613 101400      INC    0,0
06 15614 040716      STA    0,K001B
07 15615 024335      LDA    1,MXTRA
08 15616 106432      SUBZ0  0,1,SZC ;SKIP IF NEW CYL. > 76
09 15617 000726      JMP    K0011
10
11 15620 006136 KU013: RECAL
12 15621 006113      LOOP
13
14
15
16                ;LOOP TO READ A FEW SECTORS FROM THE JUST FORMATTED
17                ;DISCETTE.
18
19 15622 000403      JMP    K002
20
21 15623 000000 KU02A:  0
22 15624 000000 KU02B:  0
23
24 15625 000401 KU02:  JMP    .+1
25 15626 102520      SUBZL  0,0
26 15627 101300      MOVS   0,0
27 15630 040773      STA    0,K002A ;START WITH SECTOR 1
28 15631 102520      SUBZL  0,0
29 15632 040772      STA    0,K002B ;START WITH TRACK 1
30 15633 006112      SETP2
31 15634 006161      SETBREG
32 15635 020243      LDA    0,RDCOM
33 15636 041400      STA    0,0,3 ;SET READ COMMAND
34 15637 020764      LDA    0,K002A
35 15640 024764      LDA    1,K002B
36 15641 123000      ADD    1,0
37 15642 126520      SUBZL  1,1
38 15643 030333      LDA    2,SECSIZ
39
40 15644 006137      EXECUTE                ;EXECUTE READ COMMAND AND WAIT FOR DONE
41 15645 006162      ERROR                 ;TIME OUT
42
43 15646 101015      MOV0   0,0,SNR
44 15647 000405      JMP    .+5
45 15650 024753      LDA    1,K002A
46 15651 125300      MOVS   1,1
47 15652 030752      LDA    2,K002B
48 15653 006162      ERROR                 ;STATUS ERROR AFTER CHECKREAD WITH-
49                ;OUT DATACHECK. AC1= SECTOR NO., AC2=
50                ;TRACK NO.
51 15654 006135      RANTS                ;GET RANDOM TRACK AND SECTOR
52 15655 040746      STA    0,K002A
53 15656 044746      STA    1,K002B
54 15657 006113      LOOP
55
56                KU029:
57 15660 002177      JMP    &GRRET ;RETURN FROM DISCETTE FORMATING
```

10100 .MAIN

```
01
02 15661 054177 GR011: STA 3,GRRET
03
04 ;LOOP TO GENERATE THE PRE-WRITTEN TESTDISCETTE.
05
06 15662 000414 JMP L000
07
08 15663 000032 GENMS: 26. ;MAX SECTOR, USED IN GEN.
09 15664 000200 GENSS: 128. ;SECTOR SIZE, USED IN GEN.
10 15665 000000 L000A: 0 ;CURRENT UNIT - SET DURING INIT
11 15666 000000 L000B: 0 ;CURRENT CYL. NO.
12 15667 000000 L000C: 0 ;CURRENT HEAD NO.
13 15670 000000 L000D: 0 ;CURRENT SECTOR NO.
14 15671 052525 L000E: 052525
15 15672 006400 L000F: 128.*26. ;NO. OF DATA BYTES PER TRACK
16 15673 000000 L000G: 0 ;IF = 0 THEN WRITE MODE, ELSE CHECKREAD
17 15674 026742 L000H: TSEND+4000.
18 15675 000000 L000I: 0
19
20 15676 000401 L000: JMP .+1
21 15677 024764 LDA 1,GENMS ;MAX SECTOR
22 15700 030764 LDA 2,GENSS ;SECTOR SIZE
23 15701 006066 MULTI
24 15702 044770 STA 1,L000F
25 15703 006110 SETPO
26 15704 062677 IORST
27 15705 006061 CWAIT
28 15706 000237 FD3K0
29 15707 102440 SUBO 0,0
30 15710 040763 STA 0,L000G ;SET WRITE MODE
31
32 15711 102440 L0006: SUBO 0,0
33 15712 040754 STA 0,L000B ;CLEAR CURRENT CYL.
34 15713 102440 L0000: SUBO 0,0
35 15714 040753 STA 0,L000C ;CLEAR CURRENT HEAD
36 15715 102520 L0001: SUBZL 0,0
37 15716 040752 STA 0,L000D ;SET FIRST SECTOR
38 15717 034340 LDA 3,ATEND ;DATA ADDRESS-TRACK IMAGE IS PLACED HERE
39
40 15720 030744 L0002: LDA 2,GENSS
41 15721 151220 MOVZR 2,2
42 15722 150400 NEG 2,2
43 15723 020742 LDA 0,L000A
44 15724 041400 STA 0,0,3
45 15725 151400 INC 2,2
46 15726 175400 INC 3,3
47 15727 020737 LDA 0,L000B
48 15730 041400 STA 0,0,3
49 15731 151400 INC 2,2
50 15732 175400 INC 3,3
51 15733 020734 LDA 0,L000C
52 15734 041400 STA 0,0,3
53 15735 151400 INC 2,2
54 15736 175400 INC 3,3
55 15737 020731 LDA 0,L000D
56 15740 041400 STA 0,0,3
57 15741 151400 INC 2,2
58 15742 175400 INC 3,3
59 15743 020726 LDA 0,L000E
60 15744 041400 STA 0,0,3
61 15745 175400 INC 3,3
62 15746 151404 INC 2,2,SZR
63 15747 000775 JMP .-3
```

```

0101 .MAIN
01
02 15750 020720 LDA 0,L000D
03 15751 101400 INC 0,0
04 15752 040716 STA 0,L000D
05 15753 024710 LDA 1,GENMS
06 15754 106432 SUBZ0 0,1,SZC ;SKIP IF NEW SEC. > 26
07 15755 000743 JMP L0002
08
09 15756 006161 SETBREG
10 15757 020244 LDA 0,WRCOM
11 15760 030713 LDA 2,L000G
12 15761 151014 MOV0 2,2,SZR
13 15762 020243 LDA 0,RDCOM ;IF CHECKREAD THEN SET READ COMMAND
14 15763 041400 STA 0,0,3 ;SET WRITE COMMAND
15
16 15764 020677 LDA 0,GENMS
17 15765 101300 MOVS 0,0
18 15766 024700 LDA 1,L000B
19 15767 123000 ADD 1,0
20 15770 041401 STA 0,1,3 ;SET SEC. PER TRACK, CYL. ADDR.
21
22 15771 024676 LDA 1,L000C
23 15772 102440 SUB0 0,0
24 15773 125014 MOV0 1,1,SZR
25 15774 102620 SUBZR 0,0
26 15775 126520 SUBZL 1,1
27 15776 125300 MOVS 1,1
28 15777 123000 ADD 1,0
29 16000 024663 LDA 1,GENMS
30 16001 123000 ADD 1,0
31 16002 041402 STA 0,2,3 ;SET SURFACE,SECTOR ADDR.,NO. OF SECTORS
32
33 16003 020667 LDA 0,L000F
34 16004 041403 STA 0,3,3 ;SET NO. OF BYTES
35
36 16005 020340 LDA 0,ATEND
37 16006 030663 LDA 2,L000G
38 16007 151014 MOV0 2,2,SZR
39 16010 020664 LDA 0,L000H ;IF READ MODE THEN USE TSEND+4000
40 16011 041404 STA 0,4,3 ;SET DATA BUFFER ADDR.
41
42 16012 024266 LDA 1,CURUN
43 16013 020210 LDA 0,.1B13
44 16014 107000 ADD 0,1
45 16015 065061 DOA 1,XFDC ;START CHANNEL PROG.
46
47 16016 006145 WGSTATUS ;WAIT FOR DONE AND GET STATUS TO ACO
48 16017 006162 ERROR ;TIME OUT
49
50 16020 101015 MOV0 0,0,SNR
51 16021 000410 JMP .+8.
52 16022 034203 LDA 3,.1B5
53 16023 117414 AND0 0,3,SZR ;SKIP IF NOT WRITE PROTECT.
54 16024 000471 JMP L0005
55 16025 024641 LDA 1,L000B
56 16026 030642 LDA 2,L000D
57 16027 006162 ERROR ;STATUS ERROR AFTER WRITE. ACO= STATUS,
58 ;AC1= CYL. NO., AC2= SECTOR NO.
59 16030 000453 JMP L0004

```

!0102 .MAIN

```
01
02 16031 020642      LDA      0,L000G
03 16032 101015      MOVØ    0,0,SNR
04 16033 000421      JMP      L0007      ;IF WRITE MODE THEN GOTO NEXT
05
06 16034 024636      LDA      1,L000F
07 16035 125220      MOVZR   1,1
08 16036 124400      NEG     1,1
09 16037 044636      STA     1,L000I ;SET COUNTER
10
11 16040 030340      LDA     2,ATEND ;POINTER TO EXPECTED DATA
12 16041 034633      LDA     3,L000H ;POINTER TO RECEIVED DATA
13 16042 021400      L0008: LDA     0,0,3
14 16043 025000      LDA     1,0,2
15 16044 106415      SUBØ    0,1,SNR
16 16045 000403      JMP     .+3
17 16046 006162      ERROR   ;ERROR IN DATA WRITTEN. DISCETTE NOT OK.
18 16047 000434      JMP     L0004
19 16050 151400      INC     2,2
20 16051 175400      INC     3,3
21 16052 010623      ISZ    L000I
22 16053 000767      JMP     L0008
23
24 16054 022455      L0007: LDA     0,AL000K
25 16055 101014      MOVØ    0,0,SZR ;SKIP IF CURRENT HEAD = 0
26 16056 000407      JMP     L0003
27 16057 020271      LDA     0,MAXSU
28 16060 101015      MOVØ    0,0,SNR
29 16061 000404      JMP     L0003
30 16062 102520      SUBZL   0,0
31 16063 042446      STA     0,AL000K ;SET CURRENT HEAD = 1
32 16064 000631      JMP     L0001 ;WRITE SIDE 1
33
34 16065 102440      L0003: SUBØ    0,0
35 16066 042443      STA     0,AL000K ;SET CURRENT HEAD = 0
36 16067 022443      LDA     0,AL000L ;WRITE NEXT TRACK
37 16070 101400      INC     0,0 ;INCREMENT TO NEXT TRACK
38 16071 042441      STA     0,AL000L
39 16072 024335      LDA     1,MXTRA
40 16073 106432      SUBZØ   0,1,SZC ;SKIP IF NEXT TRACK > 76.
41 16074 000617      JMP     L0000
42 16075 022433      LDA     0,AL000J
43 16076 101014      MOVØ    0,0,SZR
44 16077 000404      JMP     L0004 ;IF CHHECKREAD THEN FINISHED.
45 16100 102000      ADC     0,0
46 16101 042427      STA     0,AL000J ;SET CHECKREAD MODE
47 16102 000607      JMP     L0006
48
49 16103 006136      L0004: RECAL
50 16104 006113      LOOP
51 16105 022341      LDA     0,IEFLG
52 16106 101014      MOVØ    0,0,SZR
53 16107 002177      JMP     AGRRET
54 16110 006043      CCRLF
55 16111 006040      CMESS
56 16112 016133      GENFI
57 16113 006043      CCRLF
58 16114 002177      JMP     AGRRET
```


!0103 .MAIN

01

02 16115 060277 L0005: INTDS

03 16116 006043 CCRLF

;WRITE PROTECT

04 16117 006046 CDICL

05 16120 006040 CMESS

06 16121 010644 DWPRO

07 16122 006044 CDISP

08 16123 010644 DWPRO

09 16124 006050 CHAAT

10 16125 006006 RESET

11 16126 006136 RECAL

12 16127 002177 JMP ;GRRRET

13

14

15 16130 015673 L000J: L000G

16 16131 015667 L000K: L000C

17 16132 015666 L000L: L000B

18

19 16133 125252 GENFI: ;TXTE !*** TESTDISCETTE OK ***!

20 120252

21 142724

22 152123

23 144504

24 141523

25 152305

26 142724

27 147640

28 120113

29 125252

30 000252

!0104 .MAIN

```
01
02          ;***** FILENAME: FDCT4
03
04 16147 000401 Z000:   JMP      .+1
05 16150 062677          IORST
06 16151 006061          CWAIT          ;IORST WAIT
07 16152 000237          F03K0
08 16153 020021x        LDA      0,POWZE
09 16154 040000          STA      0,0
10 16155 020455          LDA      0,Z000F
11 16156 040001          STA      0,1
12 16157 020452          LDA      0,Z000E
13 16160 040342          STA      0,INTAD
14 16161 020215          LDA      0,F060K
15 16162 040347          STA      0,SYSCLK ;SET SYSCLK = 4 MIN. (FOR IDU USE ONLY)
16 16163 020216          LDA      0,FD002
17 16164 061014          DOA      0,RTC ;SET CLOCKFREQ.= 100 HZ
18 16165 020435          LDA      0,Z000A
19 16166 062077          MSK0      0
20
21 16167 102000          ADC      0,0 ;CLOSE ALL SEMAPHORES
22 16170 040272          STA      0,SEMA0
23 16171 040274          STA      0,SEMA1
24 16172 040276          STA      0,SEMA2
25 16173 040300          STA      0,SEMA3
26
27 16174 042427          STA      0,AZ000B ;SET CURRENT PROCESS = -1
28
29 16175 102440          SUB0      0,0
30 16176 040273          STA      0,CLCK0 ;CLEAR ALL PROCESS CLOCKS
31 16177 040275          STA      0,CLCK1
32 16200 040277          STA      0,CLCK2
33 16201 040301          STA      0,CLCK3
34
35 16202 036422          LDA      3,AZ000C ;SET PROCESS STARTADDRESSES INTO
36                                     ;SAVE AREAS.
37 16203 031400          LDA      2,0,3
38 16204 020421          LDA      0,Z000D
39 16205 041005          STA      0,5,2
40 16206 031401          LDA      2,1,3
41 16207 020417          LDA      0,Z000D+1
42 16210 041005          STA      0,5,2
43 16211 031402          LDA      2,2,3
44 16212 020415          LDA      0,Z000D+2
45 16213 041005          STA      0,5,2
46 16214 031403          LDA      2,3,3
47 16215 020413          LDA      0,Z000D+3
48 16216 041005          STA      0,5,2
49
50 16217 060114          NIOS      RTC
51 16220 060177          INTEN
52 16221 000400          JMP      . ;WAIT FOR FIRST CLOCK INTERRUPT
53
54
55 16222 177371 Z000A: 177371
56 16223 011200 Z000B: CURPR
57 16224 011202 Z000C: PSTAB
58 16225 016372 Z000D: PRST0
59 16226 016512          PRST1
60 16227 016632          PRST2
61 16230 016752          PRST3
62 16231 011237 Z000E: MDINT
63 16232 010730 Z000F: INTSV
```

```

0105 .MAIN
01
02 ;PROCEDURE "MUEXR"
03 ;
04 ;DEFINES THE OPERATION (READ RANDOM SECTORS) AND SETS THE CHAN-
05 ;NEL PROGRAM IN MULTI-DRIVE TEST. BECAUSE THAT ALL 4 PROCESSES
06 ;WILL USE THIS PROC., IT HAS TO BE CALLED WITH INTERRUPT DISABLED
07 ;
08 ;ENTRY:          AC1= UNIT NO.
09 ;
10 ;DESTROYED:     AC0,AC1,AC2,AC3
11
12 16235 054433 MUEXR: STA 3,MUEX3
13 16234 044433 STA 1,MUEX1
14
15 16235 006135 RANTS ;GET RANDOM TRACK,SECTOR. AC0-L= SUR-
16 ;FACE+SECTOR, AC1-R= TRACK.
17
18 16236 034242 LDA 3,BASAD
19 16237 076061 DOB 3,XFDC ;SET BASE REG.
20
21 16240 030427 LDA 2,MUEX1
22 16241 157000 ADD 2,3
23 16242 035400 LDA 3,0,3
24 16243 030243 LDA 2,RDCOM
25 16244 051400 STA 2,0,3 ;SET READ COMMAND
26 16245 030334 LDA 2,MXSEC
27 16246 151300 MOVS 2,2
28 16247 147000 ADD 2,1
29 16250 045401 STA 1,1,3 ;SET SEC. PER TRACK, CYLINDER ADDRESS
30 16251 152520 SUBZL 2,2
31 16252 143000 ADD 2,0
32 16253 041402 STA 0,2,3 ;SET SURFACE,SEC.ADDR, NO. OF SECTORS.
33 16254 020333 LDA 0,SECSIZ
34 16255 041403 STA 0,3,3 ;SET NO. OF BYTES
35 16256 030262 LDA 2,RBFADR
36 16257 020410 LDA 0,MUEX1
37 16260 113000 ADD 0,2
38 16261 031000 LDA 2,0,2
39 16262 051404 STA 2,4,3 ;SET DATA BUFFER ADDRESS.
40 16263 102000 ADC 0,0
41 16264 041405 STA 0,5,3 ;INITIALIZE STATUS WORD.
42 16265 002401 JMP 4,MUEX3 ;RETURN TO PROCESS.
43
44 16266 000000 MUEX3: 0
45 16267 000000 MUEX1: 0

```

```

10106 .MAIN
01
02 ;PROCEDURE "MUExw"
03 ;
04 ;DEFINES THE OPERATION (WRITE RANDOM SECTOR) AND SETS THE CHAN-
05 ;NEL PROG. IN MULTI-DRIVE TEST.
06 ;
07 ;ENTRY:          AC1= UNIT NO
08 ;
09 ;DESTROYED:      AC0,AC1,AC2,AC3.
10
11 16270 054776 MUExw: STA      3,MUEx3
12 16271 044776      STA      1,MUEx1
13 16272 034263      LDA      3,WBFADR ;SET POINTER TO CUR. WRITE BUFFER
14 16273 137000      ADD      1,3
15 16274 035400      LDA      3,0,3
16 16275 054260      STA      3,AWBUF
17
18 16276 006142      PTFLL                      ;FILL WRITE BUFFER WITH PATTERNS
19 16277 006151      COMPT                      ;COMPLEMENT PATTERN FOR NEXT USE OF "PTFL
20 16300 034270      LDA      3,MAXDR
21 16301 175213      MOVRO  3,3,SNR ;SKIP IF 2 OR 4 DRIVES
22 16302 000403      JMP      .+3
23 16303 125015      MOVRO  1,1,SNR ;SKIP IF CURRENT DRIVE <> 0
24 16304 006151      COMPT                      ;COMPLEMENT PATTERN TWICE
25
26 16305 034262      LDA      3,RBFADR ;SET POINTER TO CUR. READ BUFFER
27 16306 137000      ADD      1,3
28 16307 035400      LDA      3,0,3
29 16310 054257      STA      3,ARBUF
30
31 16311 024232      LDA      1,FD256 ;CLEAR READ BUFFER
32 16312 124400      NEG      1,1
33 16313 102400      SUB      0,0
34 16314 041400      STA      0,0,3
35 16315 175400      INC      3,3
36 16316 125404      INC      1,1,SZR
37 16317 000775      JMP      .-3
38
39 16320 006135      RANTS                      ;GET RANDOM TRACK,SECTOR.
40
41 16321 034242      LDA      3,BASAD
42 16322 076001      DOB      3,XFDC ;SET BASE REG
43
44 16323 030744      LDA      2,MUEx1
45 16324 157000      ADD      2,3
46 16325 035400      LDA      3,0,3
47 16326 030244      LDA      2,WRCOM
48 16327 051400      STA      2,0,3 ;SET WRITE COMMAND
49 16330 030334      LDA      2,MXSEC
50 16331 151300      MOVRO  2,2
51 16332 147000      ADD      2,1
52 16333 045401      STA      1,1,3 ;SET SEC. PER TRACK, CYL. ADDR.
53 16334 152520      SUBZL   2,2
54 16335 143000      ADD      2,0
55 16336 041402      STA      0,2,3 ;SET SURFACE, SEC. ADDR, NO. OF SECTORS
56 16337 024333      LDA      1,SECSIZ
57 16340 045403      STA      1,3,3 ;SET NO OF BYTES
58 16341 024260      LDA      1,AWBUF
59 16342 045404      STA      1,4,3 ;SET WRITE BUFFER ADDRESS
60 16343 126000      ADC      1,1
61 16344 045405      STA      1,5,3 ;INITIALIZE STATUS
62 16345 002721      JMP      4,MUEx3 ;RETURN TO PROCESS

```

!0107 .MAIN

```
01
02 ;PROCEDURE "MUEXC"
03 ;
04 ;DEFINES THE OPERATION (CHECKREAD JUST WRITTEN SECTOR) AND SETS
05 ;THE CHANNELPROG. IN MULTI DRIVE TEST
06 ;
07 ;ENTRY:          AC1= UNIT NO., AC2= START OF CH.PROG SAVE AREA.
08 ;
09 ;DESTROYED:     AC0,AC2,AC3
10
11 16346 054720 MUEXC: STA      3,MUEX3
12 16347 034242      LDA      3,BASAD
13 16350 076061      DOB      3,XFDC ;SET BASE REG.
14
15 16351 137000      ADD      1,3
16 16352 035400      LDA      3,0,3
17 16353 020243      LDA      0,RDCOM
18 16354 041400      STA      0,0,3 ;SET READ COMMAND
19 16355 021000      LDA      0,0,2
20 16356 041401      STA      0,1,3 ;SET SEC. PER TRACK, CYL. ADDR.
21 16357 021001      LDA      0,1,2
22 16360 041402      STA      0,2,3 ;SET SURFACE,SEC. ADDR.,NO. OF SECTORS
23 16361 021002      LDA      0,2,2
24 16362 041403      STA      0,3,3 ;SET NO OF BYTES
25
26 16363 030262      LDA      2,RBFADR
27 16364 133000      ADD      1,2
28 16365 031000      LDA      2,0,2
29 16366 051404      STA      2,4,3 ;SET READ BUFFER ADDR.
30 16367 102000      ADC      0,0
31 16370 041405      STA      0,5,3 ;INITIALIZE STATUS WORD
32 16371 002675      JMP      4,MUEX3
```

10108 .MAIN

```
01
02           ;ALL ERRORS ON THIS PAGE REFERS TO DRIVE 0
03
04 16372 000401 PRST0:  JMP      .+1      ;PROCESS START 0
05 16373 060277          INTDS
06 16374 126440          SUB0      1,1      ;UNIT NO.
07
08 16375 006316 MEX00:  JSR      @IMUX0 ;SET CH.PROG. TO WANTED OPERATION
09
10 16376 060177          INTEN
11 16377 030506          LDA      2,MCSV0 ;SAVE COPY OF CH.PROG.
12 16400 021401          LDA      0,1,3
13 16401 041000          STA      0,0,2
14 16402 021402          LDA      0,2,3
15 16403 041001          STA      0,1,2
16 16404 021403          LDA      0,3,3
17 16405 041002          STA      0,2,2
18 16406 021404          LDA      0,4,3
19 16407 041003          STA      0,3,2
20 16410 020233          LDA      0,FD350
21 16411 040273          STA      0,CLCK0 ;SET SEMAPHORE TIMER (3.5 SEC.)
22 16412 102440          SUB0      0,0
23 16413 024210          LDA      1,.1B13
24 16414 123000          ADD      1,0
25 16415 061061          DOA      0,XFDC  ;START CHANNEL PROGRAM
26
27 16416 004451          JSR      WSEMO  ;WAIT FOR SEMAPHORE 0
28 16417 006155          ERRO0   ;TIME OUT. NO INTERRUPT FROM
29                               ;DRIVE 0 OCCURED WITHIN 3.5 SEC. AF-
30                               ;TER START.
31 16420 101014          MOV0    0,0,SZR
32 16421 006155          ERRO0   ;STATUS ERROR. STATUS <> 0. ACO= STA-
33                               ;TUS RECEIVED (IF = -1 THEN NO STATUS
34                               ;IS RECEIVED).
35
36 16422 060277          INTDS
37 16423 126440          SUB0      1,1      ;UNIT NO.
38 16424 030461          LDA      2,MCSV0
39
40 16425 000401 MEX01:  JMP      .+1      ;SET CH.PROG TO WANTED OPERATION
41
42 16426 000744          JMP      PRST0
43
44 16427 006317 MEX02:  JSR      @IMUX1 ;SET CH.PROG TO WANTED OPERATION
45
46 16430 060177          INTEN
47 16431 020233          LDA      0,FD350
48 16432 040273          STA      0,CLCK0 ;SET SEMAPHORE TIMER (3.5 SEC)
49 16433 102440          SUB0      0,0
50 16434 024210          LDA      1,.1B13
51 16435 123000          ADD      1,0
52 16436 061061          DOA      0,XFDC  ;START CHANNEL PROG.
53
54 16437 004430          JSR      WSEMO  ;WAIT FOR SEMAPHORE 0
55 16440 006155          ERRO0   ;TIME OUT
56
57 16441 101014          MOV0    0,0,SZR
58 16442 006155          ERRO0   ;STATUS ERROR
```

10109 .MAIN

```
01
02 16443 030442 LDA 2,MCSV0
03 16444 031003 LDA 2,3,2 ;AC2= WRITE BUFFER ADDR.
04 16445 034262 LDA 3,RBFADR
05 16446 126440 SUB0 1,1
06 16447 137000 ADD 1,3
07 16450 035400 LDA 3,0,3 ;AC3= READ BUFFER ADDR.
08 16451 020333 LDA 0,SECSIZ
09 16452 101220 MOVZR 0,0
10 16453 040433 STA 0,MCSV0+1 ;TEMPORARY STORE
11
12 ;COMPARE WRITTEN AND READ DATA
13 16454 021000 LDA 0,0,2
14 16455 025400 LDA 1,0,3
15 16456 106415 SUB0 0,1,SNR
16 16457 000403 JMP .+3
17 16460 006155 ERROO ;DATA READ <> DATA WRITTEN. AC0= TWO
18 ;BYTES WRITTEN,AC1= TWO BYTES READ.
19 16461 000711 JMP PRST0
20 16462 151400 INC 2,2
21 16463 175400 INC 3,3
22 16464 014422 DSZ MCSV0+1
23 16465 000767 JMP .-9. ;COMPARE NEXT TWO BYTES
24
25 16466 000704 JMP PRST0 ;REPEAT LOOP
26
27 16467 020273 WSEM0: LDA 0,CLCK0
28 16470 101015 MOV0 0,0,SNR
29 16471 001400 JMP 0,3 ;TIME OUT
30 16472 020272 LDA 0,SEMA0
31 16473 101014 MOV0 0,0,SZR
32 16474 000773 JMP WSEM0 ;WAIT ON - SEMAPHORE STILL CLOSED
33 16475 152000 ADC 2,2
34 16476 050272 STA 2,SEMA0 ;CLOSE SEMAPHORE AGAIN
35 16477 030242 LDA 2,BASAD
36 16500 102440 SUB0 0,0
37 16501 113000 ADD 0,2
38 16502 031000 LDA 2,0,2
39 16503 021003 LDA 0,5,2 ;GET STATUS
40 16504 001401 JMP 1,3
41
42
43 16505 016506 MCSV0: .+1
44 16506 000000 0
45 16507 000000 0
46 16510 000000 0
47 16511 000000 0
```

!0110 .MAIN

```
01
02          ;ALL ERRORS ON THIS PAGE REFERS TO DRIVE 1
03
04 16512 000401 PRST1:  JMP      .+1      ;PROCESS START 1
05 16513 060277          INTDS
06 16514 126520          SUBZL     1,1      ;UNIT NO.
07
08 16515 006316 MEX10:  JSR      @IMUX0 ;SET CH.PROG TO WANTED OPERATION
09
10 16516 060177          INTEN
11 16517 030506          LDA      2,MCSV1 ;SAVE COPY OF CH. PROG
12 16520 021401          LDA      0,1,3
13 16521 041000          STA      0,0,2
14 16522 021402          LDA      0,2,3
15 16523 041001          STA      0,1,2
16 16524 021403          LDA      0,3,3
17 16525 041002          STA      0,2,2
18 16526 021404          LDA      0,4,3
19 16527 041003          STA      0,3,2
20 16530 020233          LDA      0,FD350
21 16531 040275          STA      0,CLCK1 ;SET SEMAPHORE TIMER (3.5 SEC)
22 16532 102520          SUBZL     0,0
23 16533 024210          LDA      1,,1B13
24 16534 123000          ADD      1,0
25 16535 061061          DOA      0,XFDC ;START CHANNEL PROG.
26
27 16536 004451          JSR      WSEM1  ;WAIT FOR SEMAPHORE 1
28 16537 006156          ERR01    ;TIME OUT. NO INTERRUPT FROM DRIVE 1
29                                ;OCCOURED WITHIN 3.5 SEC. AFTER START
30 16540 101014          MOV@    0,0,SZR
31 16541 006156          ERR01    ;STATUS ERROR. STATUS <> 0. ACD= STATUS
32                                ;RECEIVED (IF = -1 THEN NO STATUS
33                                ;IS RECEIVED).
34
35 16542 060277          INTDS
36 16543 126520          SUBZL     1,1
37 16544 030461          LDA      2,MCSV1
38
39 16545 000401 MEX11:  JMP      .+1      ;SET CH.PROG TO WANTED OPERATION
40
41 16546 000744          JMP      PRST1
42
43 16547 006317 MEX12:  JSR      @IMUX1 ;SET CH.PROG TO WANTED OPERATION
44
45 16550 060177          INTEN
46 16551 020233          LDA      0,FD350
47 16552 040275          STA      0,CLCK1 ;SET SEMAPHORE TIMER (3.5 SEC)
48 16553 102520          SUBZL     0,0
49 16554 024210          LDA      1,,1B13
50 16555 123000          ADD      1,0
51 16556 061061          DOA      0,XFDC ;START CHANNEL PROG.
52
53 16557 004430          JSR      WSEM1  ;WAIT FOR SEMAPHORE 1
54 16560 006156          ERR01    ;TIME OUT
55
56 16561 101014          MOV@    0,0,SZR
57 16562 006156          ERR01    ;STATUS ERROR
```


!0111 .MAIN

```
01
02 16563 030442      LDA      2,MCSV1
03 16564 031003      LDA      2,3,2      ;AC2= WRITE BUFFER ADDR
04 16565 034262      LDA      3,RBFADR
05 16566 126520      SUBZL   1,1
06 16567 137000      ADD      1,3
07 16570 035400      LDA      3,0,3      ;AC3= READ BUFFER ADDR.
08 16571 020333      LDA      0,SECSIZ
09 16572 101220      MOVZR   0,0
10 16573 040433      STA      0,MCSV1+1 ;TEMPORARY SAVE
11
12                                     ;COMPARE WRITTEN AND READ DATA
13 16574 021000      LDA      0,0,2
14 16575 025400      LDA      1,0,3
15 16576 106415      SUB#    0,1,SNR
16 16577 000403      JMP      .+3
17 16600 006156      ERROR1  ;DATA READ <> DATA WRITTEN.AC0= TWO
18                                     ;BYTES WRITTEN, AC1= TWO BYTES READ.
19 16601 000711      JMP      PRST1
20 16602 151400      INC     2,2
21 16603 175400      INC     3,3
22 16604 014422      DSZ    MCSV1+1
23 16605 000767      JMP      .-9.      ;COMPARE NEXT TWO BYTES.
24
25 16606 000704      JMP      PRST1      ;REPEAT LOOP
26
27 16607 020275      WSEM1:  LDA      0,CLCK1
28 16610 101015      MOV#    0,0,SNR
29 16611 001400      JMP      0,3      ;TIME OUT
30 16612 020274      LDA      0,SEMA1
31 16613 101014      MOV#    0,0,SRZ
32 16614 000773      JMP      WSEM1      ;WAIT ON
33 16615 152000      ADC     2,2
34 16616 050274      STA      2,SEMA1 ;CLOSE SEMAPHORE AGAIN
35 16617 030242      LDA      2,BASAD
36 16620 102520      SUBZL   0,0
37 16621 113000      ADD     0,2
38 16622 031000      LDA      2,0,2
39 16623 021003      LDA      0,5,2      ;GET STATUS
40 16624 001401      JMP      1,3
41
42 16625 016626      MCSV1:  .+1
43 16626 000000      0
44 16627 000000      0
45 16630 000000      0
46 16631 000000      0
```

!0112 .MAIN

```
01
02          ;ALL ERRORS ON THIS PAGE REFERS TO DRIVE 2
03
04 16632 000401 PRST2:  JMP      .+1      ;PROCESS START 2
05 16633 060277          INTDS
06 16634 024216          LDA      1,F0002 ;UNIT NO.
07
08 16635 006316 MEX20:  JSR      4IMUX0 ;SET CH.PROG. TO WANTED OPERATION
09
10 16636 060177          INTEN
11 16637 030506          LDA      2,MCSV2 ;SAVE COPY OF CHANNEL PROG.
12 16640 021401          LDA      0,1,3
13 16641 041000          STA      0,0,2
14 16642 021402          LDA      0,2,3
15 16643 041001          STA      0,1,2
16 16644 021403          LDA      0,3,3
17 16645 041002          STA      0,2,2
18 16646 021404          LDA      0,4,3
19 16647 041003          STA      0,3,2
20 16650 020233          LDA      0,F0350
21 16651 040277          STA      0,CLCK2 ;SET SEMAPHORE TIMER (3.5 SEC)
22 16652 020216          LDA      0,F0002
23 16653 024210          LDA      1,.1B13
24 16654 123000          ADD      1,0
25 16655 061061          DOA      0,XFDC ;START CHANNEL PROG.
26
27 16656 004451          JSR      WSEM2 ;WAIT FOR SEMAPHORE 2
28 16657 006157          ERRO2   ;TIME OUT. NO INTERRUPT FROM DRIVE 2
29                                ;OCCURED WITHIN 3.5 SEC. AFTER START.
30 16660 101014          MOV#    0,0,SZR
31 16661 006157          ERRO2   ;STATUS ERROR. STATUS <> 0. ACQ= STATUS
32                                ;RECEIVED (IF = -1 THEN NO STATUS IS
33                                ;RECEIVED).
34
35 16662 060277          INTDS
36 16663 024216          LDA      1,F0002
37 16664 030461          LDA      2,MCSV2
38
39 16665 000401 MEX21:  JMP      .+1      ;SET CH.PROG. TO WANTED OPERATION
40
41 16666 000744          JMP      PRST2
42
43 16667 006317 MEX22:  JSR      4IMUX1 ;SET CH.PROG. TO WANTED OPERATION
44
45 16670 060177          INTEN
46 16671 020233          LDA      0,F0350
47 16672 040277          STA      0,CLCK2 ;SET SEMAPHORE TIMER (3.5 SEC)
48 16673 020216          LDA      0,F0002
49 16674 024210          LDA      1,.1B13
50 16675 123000          ADD      1,0
51 16676 061061          DOA      0,XFDC ;START CHANNEL PROG.
52
53 16677 004430          JSR      WSEM2 ;WAIT FOR SEMAPHORE 2
54 16700 006157          ERRO2   ;TIME OUT
55
56 16701 101014          MOV#    0,0,SZR
57 16702 006157          ERRO2   ;STATUS ERROR
```

!0113 .MAIN

```
01
02 16703 030442 LDA 2,MCSV2
03 16704 031003 LDA 2,3,2 ;AC2= WRITE BUFFER ADDR.
04 16705 034262 LDA 3,RBFADR
05 16706 024216 LDA 1,FDOO2
06 16707 137000 ADD 1,3
07 16710 035400 LDA 3,0,3 ;AC3= READ BUFFER ADDR.
08 16711 020333 LDA 0,SECSIZ
09 16712 101220 MOVZR 0,0
10 16713 040433 STA 0,MCSV2+1 ;TEMPORARY SAVE
11
12 ;COMPARE WRITTEN AND READ DATA
13 16714 021000 LDA 0,0,2
14 16715 025400 LDA 1,0,3
15 16716 106415 SUB# 0,1,SNR
16 16717 000403 JMP .+3
17 16720 006157 ERRO2 ;DATA READ <> DATA WRITTEN. AC0= TWO
;BYTES WRITTEN, AC1= TWO BYTES READ.
18
19 16721 000711 JMP PRST2
20 16722 151400 INC 2,2
21 16723 175400 INC 3,3
22 16724 014422 DSZ MCSV2+1
23 16725 000767 JMP .-9. ;COMPARE NEXT TWO BYTES
24
25 16726 000704 JMP PRST2 ;REPEAT LOOP
26
27 16727 020277 WSEM2: LDA 0,CLCK2
28 16730 101015 MOV# 0,0,SNR
29 16731 001400 JMP 0,3 ;TIME OUT
30 16732 020276 LDA 0,SEMA2
31 16733 101014 MOV# 0,0,SZR
32 16734 000773 JMP WSEM2 ;WAIT ON
33 16735 152000 ADC 2,2
34 16736 050276 STA 2,SEMA2 ;CLOSE SEMAPHORE AGAIN
35 16737 030242 LDA 2,BASAD
36 16740 020216 LDA 0,FDOO2
37 16741 113000 ADD 0,2
38 16742 031000 LDA 2,0,2
39 16743 021005 LDA 0,5,2 ;GET STATUS
40 16744 001401 JMP 1,3
41
42
43 16745 016746 MCSV2: .+1
44 16746 000000 0
45 16747 000000 0
46 16750 000000 0
47 16751 000000 0
```

10114 .MAIN

```
01
02           ;ALL ERRORS ON THIS PAGE REFERS TO DRIVE 3
03
04 16752 000401 PRST3:  JMP      .+1      ;PROCESS START 3
05 16753 060277          INTDS
06 16754 024217          LDA      1,FD003 ;UNIT NO.
07
08 16755 006316 MEX30:  JSR      4IMUX0 ;SET CH.PROG. TO WANTED OPERATION
09
10 16756 060177          INTEN
11 16757 030506          LDA      2,MCSV3 ;SAVE A COPY OF CHANNEL PROG.
12 16760 021401          LDA      0,1,3
13 16761 041000          STA      0,0,2
14 16762 021402          LDA      0,2,3
15 16763 041001          STA      0,1,2
16 16764 021403          LDA      0,3,3
17 16765 041002          STA      0,2,2
18 16766 021404          LDA      0,4,3
19 16767 041003          STA      0,3,2
20 16770 020233          LDA      0,FD350
21 16771 040301          STA      0,CLCK3 ;SET SEMAPHORE TIMER (3.5 SEC)
22 16772 020217          LDA      0,FD003
23 16773 024210          LDA      1,.1B13
24 16774 123000          ADD      1,0
25 16775 061061          DOA      0,XFDC ;START CHANNEL PROG.
26
27 16776 004451          JSR      WSEM3 ;WAIT FOR SEMAPHORE 3
28 16777 006160          ERRO3 ;TIME OUT.NO INTERRUPT FROM DRIVE 3
29                               ;OCCOURED WITHIN 3.5 SEC. AFTER START.
30 17000 101014          MOV0   0,0,SZR
31 17001 006160          ERRO3 ;STATUS ERROR. STATUS <> 0. ACO= STATUS
32                               ;RECEIVED (IF = -1 THEN NO STATUS IS
33                               ;RECEIVED).
34 17002 060277          INTDS
35 17003 024217          LDA      1,FD003
36 17004 030461          LDA      2,MCSV3
37
38 17005 000401 MEX31:  JMP      .+1      ;SET CH.PROG. TO WANTED OPERATION
39
40 17006 000744          JMP      PRST3
41
42 17007 006317 MEX32:  JSR      4IMUX1 ;SET CH.PROG. TO WANTED OPERATION
43
44 17010 060177          INTEN
45 17011 020233          LDA      0,FD350
46 17012 040301          STA      0,CLCK3 ;SET SEMAPHORE TIMER (3.5 SEC)
47 17013 020217          LDA      0,FD003
48 17014 024210          LDA      1,.1B13
49 17015 123000          ADD      1,0
50 17016 061061          DOA      0,XFDC ;START CHANNEL PROG.
51
52 17017 004430          JSR      WSEM3 ;WAIT FOR SEMAPHORE 3
53 17020 006160          ERRO3 ;TIME OUT
54
55 17021 101014          MOV0   0,0,SZR
56 17022 006160          ERRO3 ;STATUS ERROR
```

!0115 .MAIN

```
01
02 17023 030442      LDA      2,MCSV3
03 17024 031003      LDA      2,3,2      ;AC2= WRITE BUFFER ADDR.
04 17025 034262      LDA      3,RBFADR
05 17026 024217      LDA      1,F0003
06 17027 137000      ADD      1,3
07 17030 035400      LDA      3,0,3      ;AC3= READ BUFFER ADDR.
08 17031 020333      LDA      0,SECSIZ
09 17032 101220      MOVZR   0,0
10 17033 040433      STA      0,MCSV3+1 ;TEMPORARY SAVE
11
12                                     ;COMPARE WRITTEN AND READ DATA
13 17034 021000      LDA      0,0,2
14 17035 025400      LDA      1,0,3
15 17036 106415      SUB0    0,1,SNR
16 17037 000403      JMP      .+3
17 17040 006160      ERRO3   ;DATA READ <> DATA WRITTEN. ACC= TWO
18                                     ;BYTES WRITTEN, AC1= TWO BYTES READ.
19 17041 000711      JMP      PRST3
20 17042 151400      INC     2,2
21 17043 175400      INC     3,3
22 17044 014422      DSZ    MCSV3+1
23 17045 000767      JMP     .-9.      ;COMPARE NEXT TWO BYTES
24
25 17046 000704      JMP     PRST3      ;REPEAT LOOP
26
27 17047 020301      WSEM3:  LDA     0,CLCK3
28 17050 101015      MOV0    0,0,SNR
29 17051 001400      JMP     0,3      ;TIME OUT
30 17052 020300      LDA     0,SEMA3
31 17053 101014      MOV0    0,0,SZR
32 17054 000773      JMP     WSEM3    ;WAIT ON
33 17055 152000      ADC     2,2
34 17056 050300      STA     2,SEMA3  ;CLOSE SEMAPHORE AGAIN
35 17057 030242      LDA     2,BASAD
36 17060 020217      LDA     0,F0003
37 17061 113000      ADD     0,2
38 17062 031000      LDA     2,0,2
39 17063 021005      LDA     0,5,2    ;GET STATUS
40 17064 001401      JMP     1,3
41
42 17065 017066      MCSV3:  .+1
43 17066 000000      0
44 17067 000000      0
45 17070 000000      0
46 17071 000000      0
```

!0116 .MAIN

```
01
02          ;***** FILENAME: FDCT5
03
04 17072 017103 RBUFP:  RBUF0
05 17073 017503          RBUF1
06 17074 020103          RBUF2
07 17075 020503          RBUF3
08
09 17076 021103 WBUFP:  WBUF0
10 17077 021503          WBUF1
11 17100 022103          WBUF2
12 17101 022503          WBUF3
13
14 17102 000000 TSEND:  0          ;END OF TEST
15
16
17 17103 000400 RBUF0:  .BLK    256.    ;READ BUFFER 0
18
19 17503 000400 RBUF1:  .BLK    256.    ;READ BUFFER 1
20
21 20103 000400 RBUF2:  .BLK    256.    ;READ BUFFER 2
22
23 20503 000400 RBUF3:  .BLK    256.    ;READ BUFFER 3
24
25
26 21103 000400 WBUF0:  .BLK    256.    ;WRITE BUFFER 0
27
28 21503 000400 WBUF1:  .BLK    256.    ;WRITE BUFFER 1
29
30 22103 000400 WBUF2:  .BLK    256.    ;WRITE BUFFER 2
31
32 22503 000400 WBUF3:  .BLK    256.    ;WRITE BUFFER 3
33
34          .END REBIN
```

0000 SOURCE LINES IN ERROR

0117 .MAIN

A000	012540	19/03	61/09						
A001	012545	19/04	61/15						
A002	012552	19/05	61/21						
A003	012561	19/06	61/29						
A004	012570	19/07	61/38						
A005	012577	19/08	61/46						
A006	012606	19/09	61/54						
A007	012614	19/10	62/02						
A008	012622	19/11	62/09						
A009	012631	19/12	62/17						
A010	012641	19/13	62/26						
ACOMI	011560	45/41	46/05						
ADMPA	010030	12/36	28/08						
AMEXF	000302	16/02	31/22	31/23	31/24	31/25			
AMEXS	000306	16/06	31/27	31/28	31/29	31/30			
AMEXT	000312	16/10	31/32	31/33	31/34	31/35			
ANODB	007276	20/05	20/18						
APASH	010001	24/31	27/04	27/14	28/24				
ARBUF	000257	15/17	67/20	68/03	68/27	106/29			
ATBEG	000337	16/40	34/26						
ATEND	000340	16/41	34/27	77/20	92/34	96/44	100/38	101/36	
		102/11							
ATROT	007275	20/04	21/16						
AWBUF	000260	15/19	50/37	67/24	67/26	67/43	68/26	68/32	
		88/21	106/16	106/58					
AXPAS	010000	27/03	28/23						
B000	012656	19/14	63/06						
B001	012711	19/15	38/41	63/38					
B0010	012730	63/46	63/55						
B002	012733	19/16	64/04	64/08					
B002A	012732	64/06	64/11						
B003	012755	19/17	47/26	64/24	64/33				
B0030	012762	64/38	65/10						
B003A	012747	64/26	64/36	65/11					
B003B	012750	64/27	64/42						
BASAO	000242	15/02	42/17	42/47	52/35	53/16	53/52	55/20	
		56/35	57/22	58/22	59/14	59/28	63/13	65/33	
		65/42	66/17	67/38	68/05	69/22	71/16	71/36	
		72/02	72/19	72/51	75/43	78/19	87/46	96/28	
		105/18	106/41	107/12	109/35	111/35	113/35	115/35	
BASE	006040	1/01							
BASE1	006103	1/01							
BASE2	006006	1/01							
C000	013022	19/18	65/30						
CALTR	006150	12/53	84/06						
CCCOM	000251	15/10							
CCOM	000352	16/51	22/47	22/50	24/35	24/53	37/19		
CDEVI	010523	17/11	34/12	34/23	34/24				
CHACP	012024	50/36	50/43	50/50					
CHDEV	010657	12/38	36/18						
CIDUM	007534	17/13	23/02						
CIMOU	010722	36/24	36/57						
CLBPO	012267	55/21	55/27	56/02	56/14	56/37			
CLCCO	012266	55/19	56/05	56/13	56/33				
CLCKU	000273	15/36	39/10	39/13	104/30	108/21	108/48	109/27	
CLCK1	000275	15/39	39/14	39/17	104/31	110/21	110/47	111/27	
CLCK2	000277	15/42	39/18	39/22	104/32	112/21	112/47	113/27	
CLCK3	000301	15/45	39/23	39/26	104/33	114/21	114/46	115/27	
CLINS	011056	37/43	38/38						
CLKRT	011201	40/07	40/12	40/15					
CLL01	000133	12/26	12/27						
CLL02	000134	12/28	12/29						
CLL03	000135	12/30	12/31						

0118 .MAIN

CLL04	000136	12/32	12/33					
CLL05	000137	12/34	12/35					
CLL06	000140	12/36	12/37					
CLL07	000141	12/38	12/39					
CLL08	000142	12/40	12/41					
CLL09	000143	12/42	12/43					
CLL10	000144	12/44	12/45					
CLL11	000145	12/46	12/47					
CLL12	000146	12/48	12/49					
CLL13	000147	12/50	12/51					
CLL14	000150	12/52	12/53					
CLL15	000151	12/54	12/55					
CLL16	000152	12/56	12/57					
CLL17	000153	12/58	12/59					
CLL18	000154	12/60	12/61					
CLL19	000155	13/02	13/03					
CLL20	000156	13/04	13/05					
CLL21	000157	13/06	13/07					
CLL22	000160	13/08	13/09					
CLL23	000161	13/10	13/11					
CLL24	000162	13/12	13/13					
CLL25	000163	13/14	13/15					
CLRPO	012270	55/23	55/37	56/03	56/15	56/40		
CLWPO	012271	55/25	55/48	56/04	56/16	56/43		
COMIR	011302	41/32	41/41					
COMP1	006151	12/55	89/46	106/19	106/24			
CPRG0	012507	60/04	60/12					
CPRG1	012515	60/05	60/22					
CPRG2	012523	60/06	60/32					
CPRG3	012531	60/07	60/42					
CPST0	012503	15/02	60/04					
CPST1	012504	60/05						
CPST2	012505	60/06						
CPST3	012506	60/07						
CURBF	000261	15/22	53/60	57/43	88/29	90/37	91/06	
CURPR	011200	39/27	39/51	40/14	104/56			
CURUN	000266	15/30	22/02	22/04	22/09	23/26	27/21	28/16
		28/17	28/22	30/10	30/50	32/28	38/24	42/19
		42/42	47/13	52/27	53/18	53/54	53/61	57/23
		57/44	58/25	63/15	64/35	64/38	64/43	64/55
		65/06	65/12	65/35	65/43	66/19	66/37	67/17
		67/40	68/04	69/23	69/48	70/30	71/17	71/38
		71/57	72/21	72/50	75/44	77/42	77/61	85/18
		85/43	85/47	87/16	87/45	88/15	89/26	90/19
		90/23	91/26	96/30	101/42			
D000	013055	19/19	66/06	66/10				
D0000	013114	66/45	66/61					
D0001	013135	66/52	66/59	66/62				
D000A	013054	66/08	66/43	66/54	66/55	66/56		
D001	013142	19/20	67/04	67/10				
D0010	013147	67/16						
D0011	013160	67/26	68/52					
D0012	013163	67/30	67/36					
D0013	013244	68/28	68/42					
D0014	013271	68/38	68/46	68/53				
D001A	013137	67/06	67/14	68/47	68/49			
D001B	013140	67/07	68/25	68/41				
D001C	013141	67/08	67/13					
D002	013276	19/21	69/04	69/10				
D0021	013304	69/17	70/20	70/23				
D0022	013401	69/21	70/14	70/25				
D0023	013311	69/15	69/22					
D002A	013273	69/06	69/13	69/17	69/19	69/31	70/04	

0119 .MAIN

DU02B 013274	69/07	69/14	69/18	69/32	70/05	70/15	70/21
	70/22						
DU02C 013275	69/08	69/26	69/47	70/02	70/26		
DATLA 010441	33/15	33/36					
DATLO 010467	33/28	33/32	33/38				
DCURD 011736	46/52	48/39					
DEFGD 007574	23/13	23/38					
DEFID 007575	22/37	23/39					
DEFOP 010334	31/07	31/63					
DEFSL 010333	30/40	31/62					
DEV 000061	12/09	12/12					
DEVCO 010545	34/18	34/20	34/34				
DEVIC 000130	12/12	36/22	36/35	37/34	63/50		
DFRUR 010166	27/26	29/52					
DIGIN 000022 ^R XD	1/01	21/15	22/43	23/19	27/32	30/46	31/13
	32/23	32/41	33/05	33/24	34/25		
DIS 000035	1/01	45/36	45/40				
DNRUN 010120	27/25	29/11					
DOSET 012103	12/44	53/12					
DOSOU 012127	53/31	53/34					
DRONI 011247	41/07	41/11					
DR1NI 011256	41/13	41/18					
DR2NI 011265	41/20	41/25					
DR3NI 011274	41/27	41/32					
DRIDF 010331	31/60	32/17	32/35				
DUPAT 000265	15/27	55/43	55/52				
DWPRO 010644	35/41	97/23	97/25	103/06	103/08		
E000 013430	19/22	71/06	71/11				
E000A 013426	71/08	71/22					
E000B 013427	71/09	71/26					
E001 013450	19/23	71/34					
E002 013506	19/24	72/17					
E003 013705	19/27	75/05	75/11				
E003A 013702	75/07	75/25	75/53				
E003B 013703	75/08	75/22	75/48				
E003C 013704	75/09	75/54					
E004 013777	19/28	76/22	76/27				
E004A 013775	76/24	76/33					
E004B 013776	76/25	76/34					
E005 014025	19/29	77/05	77/11				
E0051 014031	77/16	77/29	78/17				
E0052 014121	77/27	78/16	78/19				
E005A 014022	77/07	77/21					
E005B 014023	77/08	77/13					
E005C 014024	77/09	77/14	77/16	77/19	78/02		
E006 014132	19/30	78/28	78/35				
E0061 014141	78/42	80/06					
E0062 014177	79/17	79/59					
E0063 014251	79/56	80/08					
E0064 014252	78/27	80/09					
E006A 014126	78/30	78/42	78/51	79/21	80/02		
E006B 014127	78/31	78/39	78/56				
E006C 014130	78/32	79/13	79/58				
E006D 014131	78/33	79/15	79/36	79/57			
E007 014252	19/31	81/05					
E008 014301	19/32	81/31	81/38				
E0081 014326	82/07	82/24					
E008A 014275	81/33	81/45					
E008B 014276	81/34	82/05					
E008C 014277	81/35	82/22	82/35				
E008D 014300	81/36	81/41	82/23				
E009 013555	19/25	73/05	73/09				
E0090 013561	73/13	73/52					

0120 .MAIN

EU091	013624	73/46	73/54					
EU09A	013554	73/07	73/12	73/21	73/41	73/47	73/49	
EU10	013627	19/26	74/06	74/10				
EU100	013633	74/15	74/60					
EU101	013700	74/54	74/62					
EU10A	013626	74/08	74/13	74/27	74/49	74/55	74/57	
EBFLG	000004A XD	1/01	16/42					
ENBSW	011567 EN	1/01	46/25					
ERRCT	000003A XD	1/01	46/07					
ERRET	000005A XD	1/01	46/06					
ERR00	006155	13/03	108/28	108/32	108/55	108/58	109/17	
ERR01	006156	13/05	110/28	110/31	110/54	110/57	111/17	
ERR02	006157	13/07	112/28	112/31	112/54	112/57	113/17	
ERR03	006160	13/09	114/28	114/31	114/53	114/56	115/17	
ERROR	006162	13/13	42/34	42/40	42/44	42/56	52/25	52/31
		59/11	59/21	61/12	61/18	61/26	61/34	61/43
		61/51	61/58	62/06	62/14	62/22	62/33	63/10
		63/26	63/30	63/42	63/45	63/52	64/15	64/17
		64/40	64/47	64/52	64/58	65/04	65/40	65/50
		65/54	66/31	66/34	66/51	67/54	67/57	68/19
		68/22	68/34	69/41	69/45	69/58	69/62	70/09
		70/36	70/40	71/14	71/24	71/28	71/52	71/55
		71/59	72/07	72/48	72/58	73/37	73/42	74/45
		74/50	75/39	75/42	75/50	75/60	76/04	76/11
		76/43	76/48	77/56	77/59	77/63	78/08	79/03
		79/06	79/23	79/30	79/38	79/49	81/19	81/23
		81/52	81/56	82/11	82/15	82/20	82/40	82/44
		83/41	83/48	84/37	84/44	85/33	85/40	85/50
		85/57	86/08	86/18	86/32	87/43	87/51	88/40
		88/47	88/61	89/10	89/17	89/38	90/46	90/49
		91/20	91/23	91/36	91/41	96/54	97/06	98/39
		98/48	99/41	99/48	101/48	101/57	102/17	
ERRTA	011561	45/54	46/06					
EXECU	006137	12/35	76/41	81/18	82/10	82/39	83/40	84/36
		85/32	88/39	88/60	89/09	90/45	91/19	98/38
		99/40						
F000	014376	19/33	83/08	83/17				
F0001	014402	83/22	84/07					
F0002	014410	83/29	83/57	83/60				
F0003	014444	83/54	84/02					
F0004	014453	84/04	84/10					
F000A	014370	83/10	83/26					
F000B	014371	83/11	83/22	83/30	83/36	83/46	84/05	
F000C	014372	83/12	83/27					
F000D	014373	83/13	83/52					
F000E	014374	83/14	83/29	83/31	83/47	83/51		
F000F	014375	83/15	83/58					
F001	014457	19/34	84/17	84/22				
F001A	014455	84/19	84/30	84/41				
F001B	014456	84/20	84/31	84/43				
F0002	000216	14/42	21/04	31/39	33/08	41/21	48/05	75/56
		81/14	81/40	104/16	112/06	112/22	112/36	112/48
		113/05	113/36					
F0003	000217	14/43	21/30	30/47	32/42	41/28	48/18	76/37
		79/46	93/37	114/06	114/22	114/35	114/47	115/05
		115/36						
F0004	000220	14/44	21/41	32/24	55/18	77/32		
F0005	000221	14/45	21/53	37/55				
F0006	000222	14/46	22/18	77/35	78/37	79/52	92/47	93/11
		93/53	94/38	94/47				
F0011	000223	14/47	93/45					
F0026	000224	14/48	57/29	72/26	73/17	73/50	87/22	93/03
		95/08	98/32					

0121 .MAIN

FD038	000225	14/49	51/23	51/39				
FD040	000226	14/50	92/40					
FD050	000227	14/51						
FD064	000230	14/52						
FD128	000231	14/53	46/58	66/40	67/27	68/24	72/34	73/27
		77/22	83/38	85/30	87/30	93/35	94/09	94/23
FD1K0	000235	14/57	28/15	33/52	95/13			
FD256	000232	14/54	55/41	55/50	58/41	58/53	94/27	106/31
FD2K0	000236	14/58	25/22	77/17				
FD350	000233	14/55	95/41	97/12	108/20	108/47	110/20	110/46
		112/20	112/46	114/20	114/45			
FD3K0	000237	14/59	28/12	33/49	38/29	62/38	63/12	63/33
		66/15	70/46	82/30	100/28	104/07		
FD400	000234	14/56	25/17					
FD5K5	000240	14/60	96/42					
FDATL	010316	31/49	33/38					
FDC	000061	12/07	12/08					
FDL0E	010326	31/57	33/18					
FDM05	000241	14/63	58/27	59/26				
FDTFL	010634	33/16	33/17	35/32				
FDTL1	010321	31/52	33/26					
FDTL2	010323	31/54	33/30					
FDTL3	010325	31/56	33/34					
FIDRV	010553	32/15	32/16	34/41				
FMSK4	011473	44/18	45/07					
FMSK5	011474	44/34	45/08					
FO010	000211	14/35	23/29	28/36	34/12	41/05	54/18	
FO014	000212	14/36	37/30					
FO077	000213	14/37	22/44	36/20	36/33	37/28	63/48	
FO377	000214	14/38	57/26	57/36	68/50	69/12	70/16	79/19
		79/27	79/34	79/42	88/54	92/42	93/06	93/48
		94/42	95/16	96/03				
FO60K	000215	14/39	24/25	36/36	104/14			
FSEC1	010320	31/51	33/25					
FSEC2	010322	31/53	33/29					
FSEC3	010324	31/55	33/33					
FSECT	010317	31/50	33/39					
FST00	000360	17/04	17/21					
FST01	000361	17/05	17/22					
FST02	000362	17/06	17/23					
FST03	000363	17/07	17/24					
FST04	000364	17/08	17/25					
FST05	000365	17/09	17/26					
FST06	000366	17/10	17/27					
FST07	000367	17/11	17/19					
FST08	000370	17/12	17/29					
FST09	000371	17/13	17/30					
FST10	000372	17/14	17/20					
FST11	000373	17/15	17/28					
FTES0	007315	17/04	20/22					
FTES1	007334	17/05	20/39					
FTES2	007361	17/06	21/02					
FTES3	007412	17/07	21/30					
FTES4	007421	17/08	21/39					
FTES5	007433	17/09	21/51					
FTES6	007462	17/10	22/16					
FTES7	007310	17/15	20/16					
FTS01	007323	20/28	20/37					
FTS10	007350	20/52	20/61					
FTS20	007367	21/09	21/13	21/14	21/20			
FTS21	007377	21/17	21/22					
FTSS0	007440	21/57	22/07	38/40				
FTS60	007472	22/25	22/27					

0122 .MAIN

FUMSK	010727	36/50	36/63					
GUOO	014513	19/35	85/08	85/16				
GUOO1	014602	86/25	86/40					
GUOO2	014620	86/37	86/42					
GUJJA	014506	85/10	85/24	85/37	86/03	86/12	86/29	
GUJUR	014507	85/11	85/27	85/39	85/54	86/30		
GUOOC	014510	85/12	86/22					
GUOOD	014511	85/13	86/23	86/39				
GUOOE	014512	85/14	86/25					
GENDA	007573	23/20	23/37					
GENDR	010152	23/11	23/12	29/39				
GENFI	016133	102/56	103/19					
GENLO	007571	23/31	23/35					
GENMS	015663	100/08	100/21	101/05	101/16	101/29		
GENNS	007572	23/21	23/36					
GENSS	015664	100/09	100/22	100/40				
GENTD	007542	17/14	23/10					
GERAN	011355	12/28	43/13					
GETTR	011433	44/26	44/33					
GRAND	006134	12/29	44/16	44/29	44/50			
GROOU	012537	20/06	61/07					
GPOU1	012655	20/07	63/04					
GROU2	013021	20/08	65/28					
GROU3	013052	20/09	66/04					
GROU4	013424	20/10	71/02					
GROU5	014366	20/11	83/02					
GROU6	014504	20/12	85/02					
GROU7	014622	20/13	87/02					
GRO10	015154	20/14	92/02					
GRO11	015661	23/35	100/02					
GRRET	000177	14/17	58/19	61/07	62/41	63/04	65/19	65/28
		65/59	66/04	70/49	71/02	82/50	83/02	84/53
		85/02	86/45	87/02	91/55	92/02	99/57	100/02
		102/53	102/58	103/12				
GTIDV	010756	37/16	37/27					
HOOO	014625	19/36	87/07	87/11				
HOOO1	014631	87/16	87/61					
HOOO2	014703	87/55	87/62					
HOOJA	014624	87/09	87/14	87/25	87/56	87/58		
HOO1	014711	19/37	88/06	88/13				
HOO11	015015	89/34	89/44					
HOO12	015030	89/22	89/40	89/47				
HOO1A	014705	88/08	88/35	88/53	89/05			
HOO1B	014706	88/09	88/32	88/44	89/14			
HOO1C	014707	88/10	88/33	88/46	89/16			
HOO1D	014710	88/11	89/25	89/43				
HOO2	015035	19/38	90/05	90/11				
HOO20	015046	90/22	90/57					
HOO21	015104	90/54	91/02					
HOO22	015106	91/05	91/49					
HOO23	015152	91/45	91/51					
HOO29	015153	90/14	91/52					
HOO2A	015032	90/07	90/38	91/12				
HOO2B	015033	90/08	90/41	91/15				
HOO2C	015034	90/09	90/18	90/26	90/39	90/52	90/56	91/03
		91/13	91/31	91/43	91/48			
HMEND	000023A XD	1/01	77/07					
IACT	010474	12/58	33/46					
IACFI	006153	12/59	20/27	20/51	21/26	21/34	21/55	22/23
		23/27						
IASOU	010220	30/12	30/30					
IASU1	010222	30/15	30/33					
IASU2	010224	30/18	30/36					

0123 .MAIN

IAS03	010244	30/21	30/53					
IAS04	010245	30/24	31/02					
IAS05	010306	30/27	31/37					
IAS06	010311	30/29	31/41					
IAS4A	010263	31/18	31/40					
IASK	010167	12/56	30/04					
IASKO	010312	30/32	30/35	30/52	30/54	31/36	31/44	
IASKR	010330	30/04	31/47	31/59	33/46	34/07		
ICHCO	012204	12/50	55/13					
IDUAD	006163	13/15	28/27	39/05				
IDUC1	007633	24/37	24/39					
IDUCT	007606	13/14	24/16					
IDUDV	007576	22/35	22/36	23/40				
IDUNK	000346	16/47	21/07	21/33	21/46	22/21	22/33	23/05
		23/25	24/20	33/57	37/21			
IDUNI	006143	12/43	42/39	52/24	71/54	77/58		
IDU01	007660	24/51	24/60	25/05				
IDU02	007657	24/38	25/04					
IDU03	007656	24/29	25/03					
IDU04	007655	24/22	25/02					
IDUPA	007664	24/27	25/10	27/34				
IEFLG	000341	16/42	66/57	68/44	70/12	73/44	74/52	78/14
		79/54	84/02	86/35	87/53	89/20	97/13	102/51
IFTS5	011060	38/30	38/40					
IGROU	007277	20/06	20/28	20/52	21/57			
IGR01	007300	20/07	20/29	20/53	21/58			
IGR02	007301	20/08	20/30	20/54	21/59			
IGR03	007302	20/09	20/31	20/55	21/60			
IGR04	007303	20/10	20/32	20/56	21/61			
IGR05	007304	20/11	20/33	20/57	21/62			
IGR06	007305	20/12	20/58					
IGR07	007306	20/13	20/34					
IGR10	007307	20/14	22/25					
ILDEV	011055	37/33	37/44	37/51	38/37			
ILICO	011057	37/53	37/56	38/39				
ILINT	011062	37/50	38/42					
ILORE	000001	XD	1/01	34/09				
IMSPR	011770	49/45	49/47					
IMUX0	000316	16/15	16/21	16/22	108/08	110/08	112/08	114/08
IMUX1	000317	16/16	16/25	16/28	108/44	110/43	112/43	114/42
IMUX2	000320	16/17	16/30					
INIAC	006147	12/51	66/13	67/12	73/13	74/15	75/13	78/44
		83/19						
INIAS	006152	12/57	20/25	20/43	21/05	21/31	21/42	21/54
		22/19						
INIMK	000131	12/14	49/28	49/43				
INI01	011764	49/30	49/42					
INISC	006154	12/61	85/19	87/17	88/16	90/20		
INSCO	010725	36/32	36/43	36/61				
INSP0	010724	36/29	36/40	36/41	36/53	36/60		
INTAD	000342	16/43	38/13	38/17	43/55	45/49	104/13	
INTAL	011061	37/39	38/41					
INTON	011025	37/36	37/41	38/10				
INTSV	010730	37/03	46/05	104/63				
IOMSK	010726	36/42	36/62					
IRESA	000020	XD	1/01					
IRETR	011015	37/54	38/02	41/41				
IRTSV	000176	14/16	37/10	38/09	38/19	39/45		
ISAV0	011053	37/06	38/07	38/35	39/35			
ISAV1	011052	37/05	38/04	38/34	39/37			
ISAV2	011051	37/04	38/03	38/12	38/33	39/39		
ISAV3	011050	37/03	38/02	38/16	38/18	38/32	39/41	
ISAVC	011054	37/08	38/05	38/10	38/36	39/43		

0124 .MAIN

ISCH	012272	12/60	56/27					
ISETP	000351	16/50	33/60	37/23				
IUNIT	012147	12/42	54/13					
IUOUT	012172	54/20	54/24	54/28	54/33			
K000	015175	19/39	92/06	92/24				
K0000	015243	93/20	95/10					
K0001	015372	95/24						
K0002	015530	95/46	97/32					
K0003	015414	95/33	95/36	95/44	97/15			
K0004	015515	97/04	97/19					
K0008	015444	95/22	96/27					
K0009	015417	92/13	92/56	93/07	93/15	93/21	93/24	93/27
		93/30	93/39	93/42	93/49	93/57	94/04	94/13
		94/18	94/43	94/51	95/17	96/02		
K000A	015156	92/08	92/55					
K000B	015157	92/09	93/20					
K000C	015160	92/10	93/41	94/17				
K000D	015161	92/11	94/03					
K000E	015164	92/14	92/29	93/23	93/33	94/07	94/35	96/22
K000F	015165	92/15	92/27	93/26	96/23			
K000G	015162	92/12	94/12					
K000H	015163	92/13	92/43	92/50				
K000I	015166	92/16	92/31	93/29	95/02	95/04		
K000J	015167	31/49	92/17	93/32	94/06	94/22		
K000K	015170	31/50	92/18	95/05	98/11			
K000L	015171	92/19	94/32	94/37				
K000M	015436	96/02	96/11	96/18	96/20	96/27	97/16	
K000N	015437	96/09	96/13	96/21				
K000O	015440	95/06	95/24	95/26	96/22	96/35	97/07	
K000P	015441	95/34	95/38	96/23	96/38			
K000Q	015442	95/29	96/24					
K000R	015443	95/42	96/25					
K000X	015177	92/26						
K000Y	015201	92/28	96/25					
K000Z	015203	92/30	96/24					
K001	015540	19/40	98/04	98/14				
K0011	015545	98/20	98/60	99/09				
K0012	015610	98/54	98/57	99/02				
K0013	015620	98/51	99/11					
K001A	015551	98/06	98/17	98/24	98/47	98/55	98/59	99/03
K001B	015532	98/07	98/18	98/27	98/30	98/46	99/04	99/06
K001C	015533	98/08						
K001D	015534	98/09	98/23					
K001E	015535	98/10	98/26					
K001F	015536	98/11	98/29					
K001G	015537	98/12	98/36	98/41				
K002	015625	19/41	99/19	99/24				
K0029	015660	97/29	99/56					
K002A	015623	99/21	99/27	99/34	99/45	99/52		
K002B	015624	99/22	99/29	99/35	99/47	99/53		
L000	015676	100/06	100/20					
L0000	015713	100/34	102/41					
L0001	015715	100/36	102/32					
L0002	015720	100/40	101/07					
L0003	016065	102/26	102/29	102/34				
L0004	016103	101/59	102/18	102/44	102/49			
L0005	016115	101/54	103/02					
L0006	015711	100/32	102/47					
L0007	016054	102/04	102/24					
L0008	016042	102/13	102/22					
L000A	015665	23/37	100/10	100/43				
L000B	015666	100/11	100/33	100/47	101/18	101/55	103/17	
L000C	015667	100/12	100/35	100/51	101/22	103/16		

0125 .MAIN

L000D	015670	100/13	100/37	100/55	101/02	101/04	101/56		
L000E	015671	100/14	100/59						
L000F	015672	100/15	100/24	101/33	102/06				
L000G	015673	100/16	100/30	101/11	101/37	102/02	103/15		
L000H	015674	100/17	101/39	102/12					
L000I	015675	100/18	102/09	102/21					
L000J	016130	102/42	102/46	103/15					
L000K	016131	102/24	102/31	102/35	103/16				
L000L	016132	102/36	102/38	103/17					
LADRA	010366	32/31	32/44	32/47					
LADRV	010566	32/33	32/34	34/53					
LARAN	011472	44/15	44/17	44/28	44/30	44/33	44/49	44/51	
		44/56	45/06						
LRCOM	000247	15/08	69/27	70/27	81/43				
LBCSV	012475	58/28	59/27	59/46					
LBSTR	012363	21/36	58/19						
LBUNI	012474	58/24	58/38	58/50	59/02	59/15	59/29	59/44	
LCCOM	000250	15/09	67/45						
L0PDS	010125	20/47	29/17						
L0PAD	010522	33/53	34/10						
LPAOR	000344	16/45	21/23	21/27					
MAXDR	000270	15/32	22/05	28/18	32/48	39/48	65/08	106/20	
MAXOP	000321	16/21	31/14						
MAXSU	000271	15/33	33/13	44/59	83/55	90/12	95/31	98/52	
		102/27							
MCSV0	016505	108/11	108/38	109/02	109/10	109/22	109/43		
MCSV1	016625	110/11	110/37	111/02	111/10	111/22	111/42		
MCSV2	016745	112/11	112/37	113/02	113/10	113/22	113/43		
MCSV3	017065	114/11	114/36	115/02	115/10	115/22	115/42		
MDINT	011237	41/03	104/62						
MERO	011645	13/02	47/31						
MEROR	011660	47/36	47/42						
MER1	011661	13/04	47/44						
MER1R	011674	47/49	47/55						
MER2	011675	13/06	48/02						
MER2R	011710	48/07	48/13						
MER3	011711	13/08	48/15						
MER3R	011724	48/20	48/26						
MERRT	011563	46/08	46/25	46/28	47/32	47/38	47/41	47/45	
		47/51	47/54	48/03	48/09	48/12	48/16	48/22	
		48/25							
MEX00	016375	16/02	108/08						
MEX01	016425	16/06	108/40						
MEX02	016427	16/10	108/44						
MEX10	016515	16/03	110/08						
MEX11	016545	16/07	110/39						
MEX12	016547	16/11	110/43						
MEX20	016635	16/04	112/08						
MEX21	016665	16/08	112/39						
MEX22	016667	16/12	112/43						
MEX30	016755	16/05	114/08						
MEX31	017005	16/09	114/38						
MEX32	017007	16/13	114/42						
MINDR	000267	15/31	22/08	27/20	28/21	30/09	32/27	32/45	
		38/23	39/30	39/50	64/37				
MOPEU	000322	16/22	32/03						
MOPE1	000325	16/25	32/04						
MOPE2	000330	16/28	32/05						
MSPRO	000013A XD	1/01	49/47						
MUEX1	016267	105/13	105/21	105/36	105/45	106/12	106/44		
MUEX3	016266	105/12	105/42	105/44	106/11	106/62	107/11	107/32	
MUEXC	016346	16/17	107/11						
MUEXR	016233	16/15	105/12						

0126 .MAIN

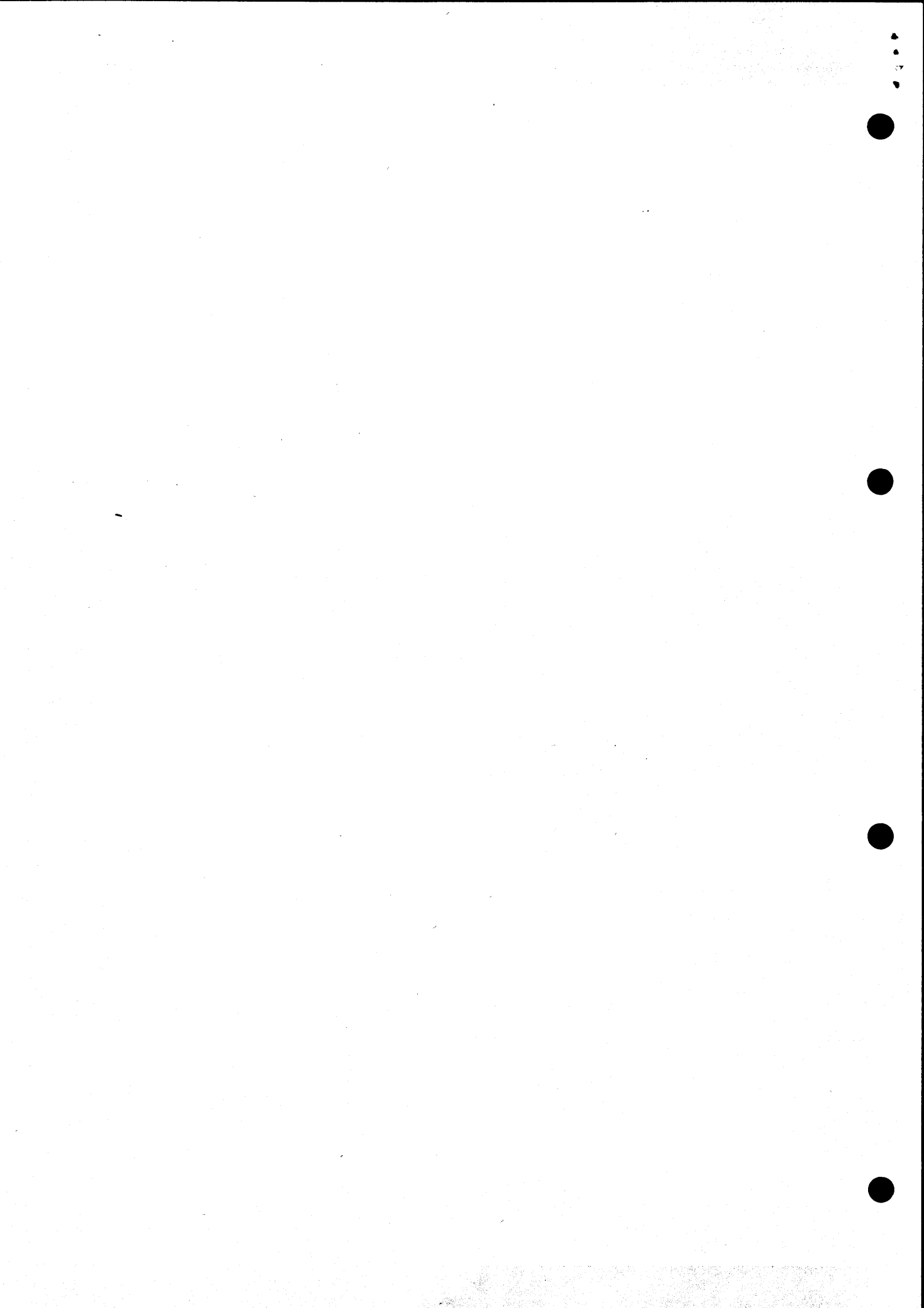
MUEXW	016270	16/16	106/11					
MULER	011504	45/60	46/09	47/37	47/50	48/08	48/21	
MULMK	000345	16/46	21/44	22/11	23/24	31/46	38/26	45/18
		45/45						
MXSEC	000334	16/37	33/41	44/20	57/31	74/21	74/58	75/18
		78/36	78/49	78/54	79/12	81/11	87/59	98/35
		98/37	98/42	105/26	106/49			
MXTRA	000335	16/38	44/41	51/30	51/42	70/19	80/04	88/56
		95/27	99/07	102/39				
NODRA	010341	30/31	30/34	31/03	31/38	31/42	32/08	32/26
NODRH	010345	20/05	32/13					
NOPAS	007777	27/02	27/33	28/32				
NOSRF	010413	23/36	30/51	32/53	33/04	33/07	33/10	
NOSUR	010601	32/58	32/59	35/02				
NWSEC	011427	44/22	44/24	44/28				
NWTRA	011452	44/43	44/45	44/49				
NXCHC	012220	55/27	56/06	56/45				
NXINS	010703	36/40	36/49	36/54				
OPPOI	010335	31/18	32/02					
PAADM	006140	12/37	20/36	20/60	22/26			
PASCO	000336	16/39						
PASET	010002	12/26	27/13					
PASFI	010061	28/29	28/36					
PASSB	000010a XD	1/01	27/04					
PASSC	000007a XD	1/01						
PASSN	000006a XD	1/01						
PASWB	012014	50/40	50/44					
PATCM	011775	12/54	50/11					
PATCO	012021	50/18	50/46					
PATFC	012007	50/34						
PATFL	012004	12/40	50/30					
PATT1	012025	50/15	50/17	50/39	50/51			
PCOM	000353	16/52	22/51	22/54	24/44			
PCOMA	007667	16/49	25/14					
POWMI	007703	24/59	26/02					
POWM1	007725	26/21	34/04					
POWM2	007752	24/50	26/43					
POWRE	000017a XD	1/01						
POWZE	000021a XD	1/01	37/11	45/43	104/08			
PRINT	000016a XD	1/01						
PROG	007200 EN	1/01	18/06					
PRST0	016572	104/58	108/04	108/42	109/19	109/25		
PRST1	016512	104/59	110/04	110/41	111/19	111/25		
PRST2	016632	104/60	112/04	112/41	113/19	113/25		
PRST3	016752	104/61	114/04	114/40	115/19	115/25		
PSTAR	011202	39/32	39/52	40/16	104/57			
PTAB0	011207	40/17	40/22					
PTAB1	011215	40/18	40/29					
PTAB2	011223	40/19	40/36					
PTAB3	011231	40/20	40/43					
PTFLL	006142	12/41	88/23	106/18				
PWCOM	000350	16/49	24/34	24/39	24/45	24/54	33/56	37/20
PWINT	011037	37/14	38/21					
RANCA	011402	43/17	43/36					
RANCB	011403	43/27	43/31	43/37				
RANCO	011404	43/18	43/21	43/38				
RANTO	011456	44/47	44/54					
RANTS	006135	12/31	84/28	85/23	88/31	99/51	105/15	106/39
RASEC	011475	44/25	44/54	45/09				
RATRA	011476	44/46	45/02	45/10				
RBCOM	000254	15/13	69/49					
RBFAD	000262	15/23	53/59	55/22	56/38	58/37	66/36	67/16
		72/36	73/29	74/36	75/31	76/05	78/58	79/10

			85/42	89/28	91/05	91/25	105/35	106/26	107/26
			109/04	111/04	113/04	115/04			
RBUF0	017103		116/04	116/17					
RBUF1	017503		116/05	116/19					
RBUF2	020103		116/06	116/21					
RBUF3	020503		116/07	116/23					
RBUFP	017072		15/22	15/23	116/04				
RCALI	011303		12/32	42/12					
RCALO	011351		42/37	42/54	42/60				
RCCOM	000255		15/14	66/22	68/10				
RDCOM	000243		15/04	72/24	73/15	74/18	75/16	76/31	81/08
			83/24	85/21	89/03	91/10	99/32	101/13	105/24
			107/17						
RDSWI	011744	EN	1/01	49/22					
REBIN	001400		1/01	116/34					
RECAL	006136		12/33	76/14	80/07	82/46	84/09	95/39	95/44
			97/10	97/28	99/11	102/49	103/11		
RECOM	000246		15/07	42/22	71/41	77/45			
RELOP	012365		58/22	59/38					
RESEC	011416		44/18	44/31					
RETRA	011434		44/34	44/52					
RETOO	000174		14/14	27/13	27/35	28/08	28/20	28/28	28/30
			28/34						
RETO1	000175		14/15	43/54	43/57				
RICOM	000253		15/12	78/47					
RNMES	010013		27/23	27/31					
RSMAS	011555		45/32	46/02					
RSPRO	010746		37/18	39/07					
RSRET	011556		45/16	45/52	45/56	45/62	46/03		
RSSVD	011557		45/17	45/50	45/59	45/61	46/04		
RSTAT	012051		12/46	52/14					
RSTIO	011477	EN	1/01	45/16					
RSTMI	011544		45/47	45/54					
RSTSK	011521		45/20	45/34					
RTC11	011155		39/31	39/47					
RTC1R	011101		37/32	39/02					
RTC0M	000256		15/15						
RTCRI	011040		38/23	39/06					
RTRSE	011411		12/30	44/13					
SAV00	000165		14/04	36/27	36/55	42/13	42/60	53/13	53/34
			55/16	56/08	56/30				
SAV01	000166		14/05	36/19	36/28	36/57	42/14	42/61	43/14
			43/32	50/12	50/31	50/46	52/16	52/42	53/14
			53/35	55/15	56/09	56/29			
SAV02	000167		14/06	36/26	36/56	42/15	42/62	43/15	43/33
			50/13	50/32	50/47	52/17	52/43	53/15	53/36
			55/14	56/10	56/28				
SAV0P	000164		14/03	36/18	36/58	37/38	42/12	42/33	42/55
			42/63	43/13	43/34	50/11	50/30	50/48	51/21
			51/46	51/47	52/15	52/30	52/34	52/44	53/12
			53/33	53/37	53/51	53/62	55/13	56/11	56/27
SAV10	000171		14/09	24/19	25/07	54/13	54/40	57/18	57/35
SAV11	000172		14/10	24/18	25/06	57/19	57/39	57/56	
SAV12	000173		14/11	24/17	25/05	44/14	45/03	54/14	54/41
			57/20	57/57					
SAV1R	000170		14/08	24/16	25/02	25/03	25/04	25/08	44/13
			45/04	57/17	57/21	57/55	57/58		
SBREG	012133		13/10	53/51					
SCHAU	012312		12/34	57/17					
SCOM	000354		16/53	22/55	22/58	24/40	33/55		
SDOUT	010720		36/46	36/55					
SECSI	000333		16/36	32/09	32/50	32/54	33/40	50/34	74/33
			75/28	76/07	76/38	79/44	81/15	82/08	82/37

		83/35	84/34	88/37	89/07	89/23	90/43	91/17
		99/38	105/33	106/56	109/08	111/08	113/08	115/08
SECSV	000356	16/55						
SELDR	010612	30/38	30/39	35/12				
SELOP	010622	31/05	31/06	35/21				
SEMAU	000272	15/35	41/36	104/22	109/30	109/34		
SEMA1	000274	15/38	41/37	104/23	111/30	111/34		
SEMA2	000276	15/41	41/38	104/24	113/30	113/34		
SEMA3	000300	15/44	41/39	104/25	115/30	115/34		
SEMBA	011275	41/04	41/35					
SERRC	011562	46/07	46/45					
SETHR	006161	13/11	73/14	74/16	75/14	76/30	78/45	81/07
		81/42	82/02	82/32	83/23	84/24	85/20	87/18
		88/25	88/50	89/02	90/32	91/08	98/20	99/31
		101/09						
SETDO	006144	12/45	63/41	64/14	64/39	65/39	71/13	
SETDV	006141	12/39	34/29					
SETIN	006146	12/49	25/20	34/06	65/16			
SETPA	006133	12/27	20/26	20/44	22/22			
SETPO	007606	16/50	24/46	24/55	25/13			
SIIDM	007475	17/12	22/30					
SIZSV	000355	16/54						
SLADR	010067	21/11	28/45					
SLOOP	000343	EN 1/01	16/44	21/25	23/23	31/45		
SPOWR	007665	25/11	25/13	25/23				
SRECI	012071	52/29	52/35					
STAD	010327	30/05	30/08	31/58				
STINT	011405	12/48	43/54					
STMSK	000264	15/26	66/32	67/55	68/20	69/43	69/60	70/38
		81/54	83/43					
STOP	0000154	XD 1/01						
STTIM	012070	52/22	52/34					
SURDF	010332	31/61	32/60					
SWISA	0000144	XD 1/01	20/20	22/60	23/07	23/33	28/42	34/32
		37/25	37/59					
SWMS1	011771	49/32	49/48					
SWMS2	011772	49/33	49/49					
SWREG	000132	12/18	23/30	28/37	34/13	46/27	49/25	
SWRET	011773	49/22	49/40	49/50				
SWSAV	011774	49/23	49/39	49/51				
SYSCL	000347	16/48	24/26	39/02	104/15			
TCOU1	012050	51/32	51/40	51/47				
TCOU2	012047	51/25	51/46					
TCURD	011725	46/50	48/29					
TFCOM	000252	15/11	96/33					
TNRUN	010110	27/24	29/02					
TRCA1	012037	51/28	51/34					
TRCA2	012044	51/37	51/42					
TRCAL	012026	12/52	51/21					
TROTB	007223	EN 1/01	19/02	20/04				
TSBEG	007274	20/02						
TSEND	017102	16/41	100/17	116/14				
UMESR	011565	46/11	46/36	46/62				
UMESU	011566	46/12	46/39	46/53	46/55			
UNMES	011573	46/36	47/14	47/35	47/48	48/06	48/19	
UNMSO	011622	46/44	46/47	46/59				
VFCOM	000245	15/06	82/03	82/33	84/25	88/51	98/21	
WBFAD	000263	15/24	55/24	56/41	58/49	67/21	87/32	88/18
		88/28	89/27	90/22	90/36	106/13		
WBUFU	021103	116/09	116/26					
WBUF1	021503	116/10	116/28					
WBUF2	022103	116/11	116/30					
WBUF3	022503	116/12	116/32					

0129 .MAIN

WRUPP 017076	15/24	116/09						
WGSTA 006145	12/47	57/54	66/30	67/53	68/18	69/40	69/57	
	70/35	73/36	74/44	75/38	79/02	81/51	96/53	
	101/47							
WORK1 000557	16/57	32/08	32/11	32/13	32/29	32/31	32/51	
	32/53	32/56	33/42					
WRCOM 000244	15/05	87/20	88/26	90/34	101/10	106/47		
WSEM0 016407	108/27	108/54	109/27	109/32				
WSEM1 016607	110/27	110/53	111/27	111/32				
WSEM2 016727	112/27	112/53	113/27	113/32				
WSEM3 017047	114/27	114/52	115/27	115/32				
XEADU 011644	47/09	47/26						
XEHAU 011636	47/11	47/16						
XEHAL 011626	13/12	47/06						
XERET 011640	47/06	47/15	47/22					
XESV0 011641	46/59	47/07	47/16	47/23	47/33	47/46	48/04	
	48/17							
XESV1 011642	46/38	46/60	47/24					
XESV2 011643	46/37	46/61	47/25					
XFDC 000061	12/08	12/09	41/03	41/09	41/15	41/22	41/29	
	42/18	42/26	42/30	42/46	45/23	45/25	45/27	
	45/29	45/30	45/31	52/21	52/39	52/40	53/17	
	53/26	53/30	53/53	54/17	57/52	58/23	59/06	
	59/10	61/56	61/57	62/05	62/12	63/09	63/14	
	63/22	63/25	63/28	63/29	64/41	64/56	64/57	
	65/02	65/34	66/18	66/28	67/39	67/51	68/06	
	68/16	69/38	69/55	70/33	71/37	71/47	71/51	
	72/20	72/43	72/47	73/34	74/42	75/36	77/31	
	77/51	77/55	78/20	78/63	81/49	87/38	87/42	
	96/29	96/51	101/45	105/19	106/42	107/13	108/25	
	108/52	110/25	110/51	112/25	112/51	114/25	114/50	
XILOR 010521	33/54	34/09						
XLORE 0000024 XD	1/01	25/28						
XPASS 0000118 XD	1/01	27/03						
XSTAA 0000124 XD	1/01	16/40						
YLOPE 007700	25/26	34/10						
Z000 016147	21/48	22/13	104/04					
Z000A 016222	104/18	104/55						
Z000B 016223	104/27	104/56						
Z000C 016224	104/35	104/57						
Z000D 016225	104/38	104/41	104/44	104/47	104/58			
Z000E 016231	104/12	104/62						
Z000F 016232	104/10	104/63						
.1B0 000200	14/22	44/57	96/40					
.1B10 000206	14/28	46/42						
.1B12 000207	14/29	72/56	82/13	82/42				
.1B13 000210	14/30	42/24	53/24	57/50	59/03	63/20	66/26	
	67/49	68/14	69/36	69/53	70/31	71/45	72/41	
	73/32	74/40	75/34	77/49	78/61	81/47	87/36	
	96/49	101/43	108/23	108/50	110/23	110/49	112/23	
	112/49	114/23	114/48					
.1B2 000201	14/23	81/21						
.1B3 000202	14/24	82/18						
.1B5 000203	14/25	97/02	101/52					
.1B6 000204	14/26	65/52						
.1B8 000205	14/27	76/46						



RETURN LETTER

Title: FDC 705 Testprogram
(Rev. 1.04)

RCSL No.: 44-RT1844

A/S Regnecentralen af 1979/RC Computer A/S maintains a continual effort to improve the quality and usefulness of its publications. To do this effectively we need user feedback, your critical evaluation of this manual.

Please comment on this manual's completeness, accuracy, organization, usability, and readability:

Do you find errors in this manual? If so, specify by page.

How can this manual be improved?

Other comments?

Name: _____ Title: _____

Company: _____

Address: _____

Date: _____

Thank you

..... Fold here

..... Do not tear - Fold here and staple

Affix
postage
here

 **REGNECENTRALEN**
af 1979

Information Department
Lautrupbjerg 1
DK-2750 Ballerup
Denmark