RL01

RL01 PERFORMANCE EXERCISER

MD-11-DZRLE-A

EP-DZRLE-A

FICHE 1 OF 1

JAN 1978

digital

MADE IN USA

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DZRLE-A-D
PRODUCT NAME:	RL01 PERFORMANCE EXERCISER
DATE CREATED:	11 OCTOBER 1977
MAINTAINER:	DIAGNOSTIC ENGINEERING
AUTHOR:	D DEKNIS

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS MANUAL.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPOSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

1.1.1 STRUCTURE OF PROGRAM

THIS DIAGNOSTIC OCCUPIES 14.5K WORDS OF MEMORY AND IS COMPATIBLE WITH BOTH XXDP AND ACT. IT CAN BE RUN STANDALONE UNDER XXDP, AND CAN BE CHAINED UNDER XXDP, ACT AND APT IN ACT MODE (SEE "CREATE CORE IMAGE" COMMAND BELOW FOR DETAILS OF CHAINING PROCEDURE). IT IS A SINGLE PROGRAM FROM THE STANDPOINT OF THE DIAGNOSTIC USER, BUT WE HAVE INCORPORATED INTO IT A CONTROL MODULE WHICH WILL LATER BE RELEASED INDEPENDENTLY AS A DIAGNOSTIC SUPERVISOR.

WHEN THIS DIAGNOSTIC IS STARTED AT ADDRESS 200, CONTROL GOES FIRST TO THE SUPERVISOR PORTION, WHICH WILL ASK CERTAIN "HARD CORE" QUESTIONS ABOUT THE ENVIRONMENT. THEN IT WILL ENTER COMMAND MODE, INDICATED BY A PROMPT CHARACTER (DS A>). AT COMMAND MODE THE OPERATOR MAY ENTER ANY OF SEVERAL COMMANDS AS DESCRIBED BELOW.

THE SUPERVISOR CODING FOLLOWS IMMEDIATELY THE DIAGNOSTIC TEST CODING, BUT THE SUPERVISOR LISTING HAS BEEN SUPPRESSED FOR GENERAL DISTRIBUTION. A LIMITED DISTRIBUTION HAS BEEN MADE TO FIELD SERVICE OF THE SUPERVISOR ASSEMBLY LISTING, AND IT MAY BE CONSULTED IN EVENT OF A SOFTWARE PROBLEM.

1.1.2 DIAGNOSTIC INFORMATION

THE RL01/RLV11 RL01 EXERCISER IS A PDP-11 (LSI-11) BASED PROGRAM. IT WILL RANDOMLY EXERCISE UP TO 2 CONTROLLERS AND 8 DRIVES. AFTER AN INITIAL WRITE OF EACH RL01, THE DRIVES ARE RANDOMLY PICKED AND GIVEN A RANDOM FUNCTION OF SEEK, GET STATUS, READ HEADER, READ OR WRITE.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

PDP-11/LSI-11 PROCESSOR WITH 16K OR MORE OF CORE
CONSOLE DEVICE (LA30,LA36,VT50,ETC.)
RL11/RLV11 CONTROLLER(S)
1 - 8 RL01 DRIVES
1 - 8 RL01K CARTRIDGES WITH BAD SECTOR FILE
KW11P, KW11L (OPTIONAL)
LINEPRINTER(OPTIONAL)

1.2.2 SOFTWARE REQUIREMENTS

MAINDEC-11-DZRLE-A

1.3 RELATED DOCUMENTS AND STANDARDS

RL01 USERS MANUAL (EK-RL01-UG-PRE)
XXDP USERS MANUAL
DIAGNOSTIC SUPERVISOR PROGRAM LISTING

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THE RL01 SUBSYSTEM SHOULD HAVE SUCCESSFULLY RUN THE FOLLOWING PROGRAMS:

MD-11-DZRLA	RL11/RLV11 RL01 CONTROLLER TEST (PART 1)
MD-11-DZRLB	RL11/RLV11 RL01 CONTROLLER TEST (PART 2)
MD-11-DVRLA	RLV11 RL01 DISKLESS TEST (RLV11 ONLY)
MD-11-DZRLC	RL01 DRIVE TEST (PART 1)
MD-11-DZRLD	RL01 DRIVE TEST (PART 2)

1.5 ASSUMPTIONS

THE HARDWARE OTHER THAN THE RL01 SUBSYSTEM IS ASSUMED TO WORK PROPERLY. FALSE ERRORS MAY BE REPORTED IF THE PROCESSOR, ETC., DO NOT FUNCTION PROPERLY.

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURES

2.1.1 LOADING PROCEDURES

FOLLOW STANDARD DEC PROCEDURES TO LOAD THE PROGRAM (XXDP, ABSOLUTE LOADER, UPD1, UPD2)

2.1.2 STARTING PROCEDURES

THE PROGRAM STARTS AT LOCATION 200. USE STANDARD DEC PROCEDURES TO START THE PROGRAM.

2.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION

THE DIAGNOSTIC CAN BE EXECUTED STANDALONE WITHOUT READING THE REMAINDER OF THIS DOCUMENT AS FOLLOWS:

A) LOAD THE DIAGNOSTIC
B) START AT ADDRESS 200
C) ANSWER THE HARDCORE QUESTIONS
D) RECEIVE PROMPT (DS A>)
E) ENTER STA<CR>
F) ANSWER HARDWARE AND SOFTWARE QUESTIONS
G) GET END OF PASS MESSAGES OR ERROR MESSAGES
H) TO END EXECUTION, ENTER CONTROL/C

2.2 SPECIAL ENVIRONMENTS

THE ENVIRONMENTS THIS PROGRAM WILL RUN IN ARE XXDP, XXDP CHAIN, ACT, SLIDE AND APT.

2.3 PROGRAM OPTIONS

2.3.1 START COMMAND

```
*************************************************************************
STA(RT)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:<FLAG-LIST>/EOP:<INCR>
*************************************************************************
```

2.3.1.1 TESTS SWITCH (/TESTS:<TEST-LIST>)

<TEST-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (1:2 ETC.) OR RANGES OF DECIMAL NUMBERS (1-5:8-10 ETC.) SEPARATED BY COLONS, SPECIFYING WHICH TESTS IT IS DESIRED BE EXECUTED. THE TEST NUMBERS RANGE FROM 1 TO THE LARGEST TEST NUMBER IN THE DIAGNOSTIC. THEY MAY BE SPECIFIED IN ANY ORDER. TESTS WILL BE EXECUTED IN NUMERICAL ORDER REGARDLESS OF ORDER OF SPECIFICATION. THE DEFAULT IS TO EXECUTE ALL TESTS. ON THIS AND ALL SWITCHES, THE ANGLE BRACKETS <> ARE PUNCTUATION USED IN THE DEFINITION ONLY, AND ARE NOT TO BE TYPED BY THE OPERATOR. SEE EXAMPLE AT END OF 2.3.1.

2.3.1.2 PASS SWITCH (/PASS:<PASS-CNT>)

<PASS-CNT> IS A DECIMAL NUMBER INDICATING THE DESIRED NUMBER OF PASSES. A PASS IS DEFINED AS THE EXECUTION OF THE FULL DIAGNOSTIC (ALL SELECTED TESTS) AGAINST ALL UNITS SUBMITTED. THE DEFAULT IS NON-ENDING EXECUTION: IE, EXIT IS ACCOMPLISHED EITHER BY TYPING A CONTROL/C OR BY A HALT ON ERROR BEING ENCOUNTERED, IN WHICH CASE WE RETURN TO COMMAND MODE. SEE EXAMPLE AT END OF 2.3.1.

2.3.1.3 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS A SEQUENCE OF ELEMENTS OF THE FORM <FLAG>, <FLAG=1>, OR <FLAG=0>, SEPARATED BY COLONS, WHERE <FLAG> HAS ONE OF THE FOLLOWING VALUES:

```
  HOE     HALT ON ERROR. CAUSING COMMAND MODE TO BE ENTERED WHEN AN
          ERROR IS ENCOUNTERED
  LOE     LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY
          WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT,
          SUBTEST, OR TEST) CONTAINING THE ERROR
  IER     INHIBIT ERROR REPORTING
  IBE     INHIBIT BASIC ERROR REPORTS
  IXE     INHIBIT EXTENDED ERROR REPORTS
  PRI     DIRECT ALL MESSAGES TO A LINE PRINTER
  PNT     PRINT NUMBER OF TEST BEING EXECUTED
  BOE     BELL ON ERROR
  UAM     RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS
  ISR     INHIBIT STATISTICAL REPORTS
  IDR     INHIBIT DROPPING OF UNITS BY DIAGNOSTIC
```

THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED. SEE EXAMPLE AT END OF 2.3.1.

2.3.1.4 END OF PASS SWITCH (/EOP:<INCR>)

<INCR> IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE PRINTED. THE DEFAULT IS AT THE END OF EVERY PASS. SEE EXAMPLE AT END OF 2.3.1.

2.3.1.5 EFFECT OF COMMAND

THE EFFECT OF THE START COMMAND IS TO INITIATE THE HARDWARE PARAMETER DIALOGUE, THE SOFTWARE PARAMETER DIALOGUE, AND THEN THE DIAGNOSTIC TESTS THEMSELVES.

THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE QUESTION "# UNITS?" TO WHICH THE OPERATOR REPLIES WITH A DECIMAL NUMBER N FROM 1 TO 64. THE TERM "UNIT" REFERS TO THE DEVICE TO WHICH THIS SERIES OF DIAGNOSTICS IS DEDICATED. FOLLOWING THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES WILL BE BUILT. EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING ALL THE HARDWARE INFORMATION FOR ONE UNIT. THE OPERATOR MUST SUPPLY N (NUMBER OF UNITS) VALUES FOR EACH QUESTION. HE MAY DO THIS BY GIVING ONE ANSWER TO EACH QUESTION (IN WHICH CASE THE SERIES OF QUESTIONS WILL BE POSED N TIMES) OR BY GIVING N VALUES, SEPARATED BY COMMAS, TO EACH QUESTION (SERIES WILL BE POSED ONCE). EACH QUESTION IS FOLLOWED BY THE RESPONSE RADIX (D FOR DECIMAL, B FOR BINARY, O FOR OCTAL, L FOR YES/NO) IN PARENTHESES AND THE DEFAULT VALUE AFTER THE PARENTHESES.

FOLLOWING THE HARDWARE QUESTIONS ARE THE SOFTWARE QUESTIONS TO BUILD THE SOFTWARE TABLES, WHICH DEFINE THE MODE (QUICK VERIFY ETC.) THAT THE DIAGNOSTIC WILL EXECUTE IN.

AT THE POINT WHERE THE QUESTION "# UNITS?" IS ANSWERED, CORE STORAGE IS ALLOCATED FOR THE P-TABLES, AND IF THERE IS NOT ENOUGH TO ACCOMMODATE THEM THE MESSAGE "TOO MANY UNITS" IS ISSUED. IN THIS CASE THE DIAGNOSTIC MUST BE EXECUTED MORE THAN ONCE TO TEST ALL UNITS.

EXAMPLE:

```
STA/TESTS:1:2-4:6:8-10/PASS:3/FLAGS:IER:HOE=1:UAM:LOE
```

THIS COMMAND WILL CAUSE THREE PASSES TO BE MADE, EACH PASS CONSISTING OF TESTS 1,2,3,4,6,8,9, AND 10 EXECUTED AGAINST ALL UNITS. THERE IS NO DIFFERENCE BETWEEN SAYING <FLAG> AND SAYING <FLAG=1>. THE NOTATION <FLAG=0> IS MEANINGFUL ONLY ON A COMMAND OTHER THAN START TO CLEAR A FLAG THAT WAS PREVIOUSLY SET. NOTE THAT ON ALL COMMANDS ONLY THE FIRST THREE LETTERS ARE SCANNED.

2.3.2 RESTART COMMAND

```
****************************************************************************
RES(TART)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:<FLAG-LIST>/UNITS:<UNIT-LIST>
****************************************************************************
```

2. 3. 2. 1 TESTS, PASS, AND FLAGS SWITCHES

<TEST-LIST>, <PASS-CNT>, AND <FLAG-LIST> ARE AS IN THE START COMMAND.

2. 3. 2. 2 UNITS SWITCH (/UNITS: <UNIT-LIST>)

<UNIT-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (1,2 ETC.) OR RANGES OF DECIMAL NUMBERS (1-5, 8-10 ETC.) SEPARATED BY COLONS, INDICATING WHICH UNITS IT IS DESIRED BE TESTED. THE NUMBERS MAY RANGE FROM 1 THRU N (N IS THE NUMBER OF UNITS SPECIFIED IN THE PREVIOUS START COMMAND). THE NUMBER INDICATES THE POSITION OF THE P-TABLE AS THE DATA WAS ENTERED DURING THE HARDWARE DIAGLOGUE. THE UNITS WHICH ARE SELECTED MUST NOT HAVE BEEN DROPPED BY THE DROP COMMAND. SEE THE DISCUSSION OF ADD AND DROP COMMANDS BELOW. DEFAULT IS TO TEST ALL UNITS WHICH HAVE NOT BEEN DROPPED BY A DROP COMMAND.

2. 3. 2. 3 EFFECT OF COMMAND

THE RESTART COMMAND DIFFERS FROM THE START COMMAND IN THAT THE P-TABLES FROM THE PREVIOUS START COMMAND (THERE MUST HAVE BEEN ONE) ARE USED, INSTEAD OF NEW ONES BEING BUILT. THE UNITS SWITCH GIVES THE ABILITY TO SELECT A SUBSET OF THESE. THE SOFTWARE DIALOGUE MAY OPTIONALLY BE RE-EXECUTED (OPERATOR WILL BE ASKED). THE COMMAND CAN BE USED AFTER COMMAND MODE HAS BEEN REENTERED IN ANY OF THE THREE NORMAL WAYS: A) THE REQUESTED NUMBER OF PASSES HAVE BEEN MADE B) AN ERROR WAS ENCOUNTERED WITH THE HALT ON ERROR FLAG SET C) A CONTROL/C WAS ENTERED BY THE OPERATOR.

2. 3. 3 CONTINUE COMMAND

```
*******************************************
CON(TINUE)/PASS: <PASS-CNT/FLAGS: <FLAG-LIST>
*******************************************
```

2. 3. 3. 1 PASS SWITCH (/PASS: <PASS-CNT>)

<PASS-CNT> IS SAME AS IN START COMMAND, BUT THE DEFAULT IS THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART. IF NONE REMAINS, THE DEFAULT IS NON-ENDING EXECUTION.

2. 3. 3. 2 FLAG SWITCH (/FLAGS: <FLAG-LIST>)

<FLAG-LIST> IS SAME AS IN START COMMAND BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

2. 3. 3. 3 EFFECT OF COMMAND

CONTINUE MUST FOLLOW A START OR RESTART, AND COMMAND MODE MUST HAVE BEEN ENTERED DUE TO A HALT ON ERROR OR A CONTROL/C. THE EFFECT OF THE COMMAND IS TO GO TO THE BEGINNING OF THE TEST THAT WAS BEING EXECUTED WHEN THE HALT OR CONTROL/C TOOK PLACE. SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED. HARDWARE PARAMETERS MAY NOT BE CHANGED

2. 3. 4 PROCEED COMMAND

```
****************************
```

PRO(CEED)/FLAGS: <FLAG-LIST>
****

2.3.4.1 FLAGS SWITCH (/FLAGS: <FLAG-LIST>)

<FLAG-LIST> IS AS IN THE START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

2.3.4.2 EFFECT OF COMMAND

PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE. COMMAND MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR. THE EFFECT OF THE COMMAND IS TO BEGIN EXECUTION AT THE LOCATION FOLLOWING THE ERROR CALL. NEITHER HARDWARE NOR SOFTWARE PARAMETERS MAY BE ALTERED.

2.3.5 CREATE CORE IMAGE COMMAND

****
CCI/TESTS: <TEST-LIST>/PASS: <PASS-CNT>/FLAGS: <FLAG-LIST>
****

2.3.5.1 TESTS, PASS, AND FLAGS SWITCHES

<TEST-LIST>, <PASS-CNT>, <FLAG-LIST>, AND ARE AS IN THE START COMMAND, EXCEPT THAT THE UAM (UNATTENDED MODE) FLAG DEFAULTS TO THE SET POSITION.

2.3.5.2 EFFECT OF COMMAND

THE PURPOSE OF THIS COMMAND IS TO CREATE A BIC FILE SUITABLE FOR CHAIN MODE EXECUTION. THE XXDP PROCEDURE IS AS FOLLOWS:

```
INVOKE THE XXDP UTILITY UPD1
LOAD XXN:FILE.BIN,
START 200
<QUESTIONS AND ANSWERS>
RESTART UPD1 USING RESTART ADDRESS
HICORE ADDRESS (IF "PASSED 14.5K" MESSAGE CAME)
DUMP XXN:FILE.BIC
```

THE OPERATOR DIALOGUE (HARDWARE AND SOFTWARE) WILL BE EXECUTED AS IN THE START COMMAND; BUT AT THE END OF THE QUESTIONS THE HALT STATE WILL BE ENTERED, AT WHICH TIME THE OPERATOR SHOULD DUMP THE PROGRAM TO THE XXDP LIBRARY USING A BIC EXTENSION TO INDICATE THAT THIS FILE IS CHAINABLE. HE SHOULD USE THE XXDP UTILITY "UPD1" TO DO THIS. IF THE P-TABLES EXTEND BEYOND 14.5K, A MESSAGE WILL BE ISSUED GIVING THE NEW UPPER CORE LIMIT, TO WHICH THE OPERATOR MUST ADJUST BEFORE DUMPING. HE MAY NOW DELETE THE NON-CHAINABLE BIN FILE IF DESIRED, SINCE THE BIC FILE HAS ALL THE CAPABILITIES OF IT.

WHEN THIS BIC FILE IS SUBSEQUENTLY EXECUTED IN CHAIN MODE, THE OPERATOR DIALOGUES WILL BE BYPASSED. HOWEVER, IF IT IS EXECUTED STANDALONE, THE DIALOGUE WILL BE REISSUED.

NOTE THAT IF THE MESSAGE "TOO MANY UNITS" IS ISSUED, TWO OR MORE CORE IMAGES MUST BE CREATED (WITH DIFFERENT NAMES) TO TEST ALL UNITS.

```
2.3.6 ADD COMMAND
      -----------

*********************
ADD/UNITS:<UNIT-LIST>
*********************

2.3.6.1 UNITS SWITCH  (/UNITS:<UNIT-LIST)

        <UNIT-LIST> IS AS IN THE RESTART COMMAND.

2.3.6.2 EFFECT OF COMMAND

        THE UNITS SPECIFIED ARE ADDED TO THE TEST SEQUENCE.  EACH UNIT
        MUST HAVE A P-TABLE IN MEMORY DUE TO AN EARLIER HARDWARE DIALOGUE.
        THIS COMMAND MUST BE FOLLOWED BY A RESTART OR CONTINUE.  THE UNITS
        SWITCH MUST BE SPECIFIED.  THE ADD COMMAND IS MEANINGFUL ONLY FOR UNITS
        THAT WERE PREVIOUSLY DROPPED.

2.3.7 DROP COMMAND
      ------------

************************
DRO(P)/UNITS:<UNIT-LIST>
************************

2.3.7.1 UNITS SWITCH  (/UNITS:<UNIT-LIST>)

        <UNIT-LIST> IS AS IN THE RESTART COMMAND.

2.3.7.2 EFFECT OF COMMAND

        THE UNITS SPECIFIED WILL BE DROPPED FROM TESTING.  THE UNITS WILL
        BE RESELECTED ONLY BY THE EXECUTION OF AN ADD OR START COMMAND.
        THE UNITS SWITCH MUST BE ENTERED.  THIS COMMAND MUST BE
        FOLLOWED BY A RESTART OR A CONTINUE COMMAND.

2.3.8 PRINT COMMAND
      -------------

*******
PRI(NT)
*******

2.3.8.1  EFFECT OF COMMAND

        ALL STATISTICS TABLES ACCUMULATED BY THE DIAGNOSTIC ARE PRINTED.  THE
        ISR (INHIBIT STATISTICAL REPORTING) FLAG IS CLEARED.

2.3.9 DISPLAY COMMAND
      ---------------

***************************
DIS(PLAY)/UNITS:<UNIT-LIST>
***************************

2.3.9.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)

        <UNIT-LIST> IS AS IN THE RESTART COMMAND.
```

2.3.9.2 EFFECT OF COMMAND

THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED OUT IN THE FORMAT IN WHICH THEY WERE ENTERED. ANY UNITS THAT WERE DROPPED BY THE OPERATOR "DROP" COMMAND ARE SO DESIGNATED.

2.3.10 FLAGS COMMAND

```
*******
FLA(GS)
*******
```

2.3.10.1 EFFECT OF COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

2.3.11 ZFLAGS COMMAND

```
********
ZFL(AGS)
********
```

2.3.11.1 EFFECT OF COMMAND

ALL FLAGS ARE CLEARED.

2.3.12 HARDWARE PARAMETERS

THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND. THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONCE.

RL11 (L) Y?

ANSWER YES(Y) IF YOU HAVE AN RL11 CONTROLLER, NO(N) IF YOU HAVE AN RLV11 CONTROLLER.

BUS ADDRESS (O) 174400?

ANSWER WITH THE BUS ADDRESS OF THE CONTROLLER.

VECTOR (O) 330?

ANSWER WITH THE INTERRUPT VECTOR OF THE CONTROLLER.

BR LEVEL (O) 5?

ANSWER WITH THE INTERRUPT PRIORITY OF THE CONTROLLER.

DRIVE (O) 0?

ANSWER WITH THE DRIVE(S) CONNECTED TO THE CONTROLLER.

2.3.13 SOFTWARE PARAMETERS

THE FOLLOWING QUESTIONS ARE ASKED IF REQUESTED ON A START, RESTART, OR CONTINUE. THEY ALLOW FLEXABILITY IN THE WAY THE PROGRAM BEHAVES. THE SOFTWARE PARAMETERS GIVE THE PROGRAM FLEXIBILITY IN THE WAY IT RUNS. THE PARAMETERS CAN BE MODIFIED ON A START, RESTART, OR CONTINUE BY ANSWERING (Y)ES TO THE FOLLOWING QUESTION:

CHANGE S.W. ?

A YES ANSWER WILL ASK THE FOLLOWING SOFTWARE PARAMETER QUESTIONS, WITH THE PRESENT DEFAULT VALUE PRINTED TO THE LEFT OF THE QUESTION MARK. (THE LAST ANSWER GIVEN IS THE DEFAULT) THE DEFAULT IS TAKEN ON A <CR>. CONTROL Z (Z) WILL DEFAULT ALL REMAINING QUESTIONS AND START THE TEST.

2. 3. 13. 1 RETRY LMT X?

THIS IS THE NUMBER OF TIMES THE PROGRAM WILL ATTEMPT A COMMAND BEFORE IT QUITS AND REPORTS A HARD ERROR. IF THE RETRY IS SUCCESSFUL BEFORE THE RETRY LIMIT IS EXCEEDED IT WILL PRINT AND LOG A SOFT ERROR.

LIMITS 0 - 65,535

2. 3. 13. 2 SEEK RETRY LMT X?

THIS IS THE NUMBER OF RETRYS THAT WILL BE ATTENPTED TO SEEK TO A CYLINDER ON A MIS-SEEK. AFTER RETRY IS EXHAUSTED, WE WILL NOT TRY FOR THAT CYLINDER BUT CONTINUE WITH A NEW CYLINDER.

LIMITS 0 - 65,535

2. 3. 13. 3 DATA DMP ON DCK ERR X?

GIVES THE ABILITY TO SEE THE 1 SECTOR BUFFER THAT HAD A DATA CRC ERROR. THE RESULTS OF THE PRINTOUT ARE ONE OF TWO POSSIBILTIES.

1) ONLY THOSE WORDS OF THE SECTOR THAT WERE BAD
   ARE PRINTED WITH WHAT WAS EXPECTED.
2) IF ONE OF THE 1ST TWO WORDS IS BAD
   (USED TO KEY) THE ENTIRE BUFFER IS DUMPED.

LIMITS Y OR N

2. 3. 13. 4 # OF ERR DUMPED

THIS IS THE NUMBER OF MISCOMPARES THAT WILL BE PRINTED.

LIMITS 0 - 128

2. 3. 13. 5 TIME BETW REPORTS (MIN) X?

THIS IS THE INTERNAL BETWEEN AUTOMATIC STATISTIC REPORTS ON ALL DRIVES IF A CLOCK IS PRESENT AND WAS ANSWERED SO IN THE INITIAL DIALOG.

LIMITS 1 - 65,535

2.3.13.6 DROP DR ON ERR LMTS REACHED X?

GIVES THE ABILITY TO AUTOMATICALLY STOP TESTING ON A DRIVE ONCE ONE OF THE ERROR LIMITS HAVE BEEN EXCEEDED (SEEK, DRIVE, HARD, SOFT). IF THE ANSWER IS YES THEN THE FOLLOWING FOUR QUESTIONS WILL BE ASKED, IF NO THEN THE NEXT QUESTION WILL BE 2.3.13.11.

LIMITS Y OR N

2.3.13.7 HRD ERR LMT X?

THIS IS THE LIMIT OF HARD ERRORS THAT A DRIVE WILL BE DROPPED ON. A HARD ERROR IS ONE ON WHICH THE RETRY HAS BEEN EXHAUSTED.

LIMITS 1 - 65,535

2.3.13.8 SFT ERR LMT X?

THIS IS THE LIMIT OF SOFT ERRORS THAT A DRIVE WILL BE DROPPED ON. A SOFT ERROR IS AN ERROR ON AN OPERATION THAT WAS SUCCESSFUL WITHIN THE RETRY LIMIT.

LIMITS 1 - 65,535

2.3.13.9 DATA MISCOMPARE LIMIT X?

THIS IS THE LIMIT OF IN CORE MISCOMPARES THAT THE DRIVE WILL BE DROPPED ON.

LIMITS 1 - 65,535

2.3.13.10 SK ERR LMT X?

THIS IS THE LIMIT OF MIS-SEEK AND TRACKING ERRORS THAT A DRIVE WILL BE DROPPED ON

LIMITS 1 - 65,535

2.3.13.11 DR ERR LMT X?

THIS IS THE LIMIT OF DRIVE ERRORS THAT A DRIVE WILL BE DROPPED ON.

LIMITS 1 - 65,535

2.3.13.12 DROP DR ON OPER LMTS REACHED X?

GIVES THE ABILITY TO STOP TESTING ON A DRIVE THAT HAS EXCEEDED CERTAIN OPERATION LIMITS (SEEK, BITS TRANSFERRED). THE DRIVE WILL BE DROPPED ONLY WHEN BOTH HAVE BEEN EXCEEDED. IF THE ANSWER IS YES THEN THE NEXT TWO QUESTIONS WILL BE ASKED, IF NO THE NEXT QUESTION WILL BE 2. 3. 13. 15.

LIMITS Y OR N

2. 3. 13. 13 DATA XFER LMT (*10(10)) X?

THIS IS THE LIMIT OF COMBINED BITS READ/WRITTEN (*10(10)) ON WHICH THE DRIVE WILL BE DROPPED.

LIMITS 1 - 65,535

2. 3. 13. 14 SK LMT (*10(3)) X?

THIS IS THE LIMIT OF SEEK OPERATIONS (*10(3)) ON WHICH THE DRIVE WILL BE DROPPED.

LIMITS 1 - 65,535   (*10(3))

2. 3. 13. 15 DO YOU WANT TO CHANGE SEEK, R/W PARAMETERS X?

THE NORMAL OPERTATION IS TO SEEK AND TRANSFER ON THE ENTIRE CARTRIDGE, CYLINDERS 0 - 255, SECTORS 0 - 39 AND BOTH SURFACES. THE NORMAL TRANSFER IS RANDOM BETWEEN 3 AND 1280 WORDS.

THE NEXT 8 PARAMETERS WILL ALLOW THE USER TO CONFINE THE TESTING TO ANY CONTIGUOUS SECTION OF THE CARTRIDGE AND CONTROL THE SIZE OF THE TRANSFERS.

A YES ANSWER WILL ASK THE NEXT 13 QUESTIONS, A NO WILL GO TO QUESTION 2. 3. 13. 29.

2. 3. 13. 16 STIPULATE R/W XFER SIZE X?

THE PROGRAM WILL NORMALLY MAXIMIZE THE TRANSFER SIZE BY USING ALL OF MEMORY (<28K) AVAILABLE. THIS QUESTION IF ANSWERED YES WILL RESTRICT THE BUFFER TO THOSE VALUES GIVEN IN 2. 3. 13. 17 AND 2. 3. 13. 18. IF NO IS GIVEN NEXT QUESTION IS 2. 3. 13. 19.

LIMITS Y OR N

2. 3. 13. 17 MAX XFER X?

REPRESENTS THE MAXIMUM AMOUNT OF WORDS TO READ OR WRITE

LIMITS 3 - 5120

```
2. 3. 13. 18 MIN XFER X?

        REPRESENTS THE MINIMUM AMOUNT OF WORDS TO READ OR WRITE

                LIMITS 3 - 5120

2. 3. 13. 19 RD ONLY X?

        GIVES THE ABILITY TO INHIBIT WRITING THE PACK WHILE
        TESTING, THE INITIAL WRITE OF THE PACK FROM THE
        START COMMAND WILL STILL OCCUR.

                LIMITS Y OR N

2. 3. 13. 19 RAN PAT X?

        NORMAL OPERATION SHOULD BE YES, BUT THIS PARAMETER
        WILL ALLOW THE WRITING OF ONLY ONE PATTERN OF
        EIGHT NORMAL PATTERNS.  THE PATTERNS ARE
        SHOWN IN 2. 3. 13. 21.  IF ANSWER IS YES THEN 2. 3. 13. 21
        WILL BE THE NEXT QUESTION, IF NO THEN QUESTION 2. 3. 13. 22.

                LIMITS Y OR N

2. 3. 13. 21 WHICH ONE X?

        IT IS NOW POSSIBLE TO CONTAIN THE EXERCISER IN
        WRITING ONLY ONE OF THE FOLLOWING EIGHT PATTERNS:

        0 -     ALL 0'S
        1 -     177777, 177777, 177777, 52525, 52525, 52525
                177777, 177777, 52525, 52525, 177777, 52525
                177252, 177252, 172765, 172765
        2 -     0, 0, 0, 177777, 177777, 177777
                0, 0, 177777, 177777, 0, 177777, 0, 177777
                0, 177777
        3 -     25252, 52525, 52525, 125252, 125252, 125252
                52525, 52525, 125252, 125252, 52525, 125252
                52525, 125252, 52525, 125252
        4 -     155555, 133333, 66666, 155555, 133333, 66666
                155555, 133333, 66666, 155555, 133333, 66666
                155555, 133333, 66666, 155555
        5 -     121105, 150442, 64221, 132110, 55044, 26422
                13211, 105504, 42642, 21321, 110550, 44264
                22132, 11055, 104426, 42213
        6 -     ALL 1'S
        7 -     45513, 122645, 151322, 64551, 132264, 55132
                26455, 113226, 45513, 122645, 151322, 64551
                132264, 55132, 26455, 113226

                LIMITS 0 - 7

2. 3. 13. 22  WR CHK X?

        DO YOU WISH TO PERFORM A WRITE CHECK AFTER EACH WRITE OPERATION

                LIMITS Y OR N
```

NORMAL TRANSFERS ARE RANDOM BETWEEN 3 AND 1280 WORDS. THIS PARAMETER WILL ALLOW YOU TO SPECIFY HOW MANY WORDS SHOULD BE COMPARED PER SECTOR IN CORE AFTER EACH READ. IF THE VALUE SPECIFIED IS GREATER THAN THAT READ IN ONLY THE NUMBER READ IN ARE COMPARED. THE FEWER WORDS COMPARED IN CORE ON EACH READ THE FASTER THROUGHPUT THE EXERCISER WILL HAVE.

LIMITS 0 - 128

2.3.13.25 # OF DATA ERR RPT'D PER BUF X?

THIS PARAMETER WILL LIMIT THE NUMBER OF IN CORE MISCOMPARES PRINTED. THE PROGRAM WILL CONTINUE TO COMPARE AS MANY WORDS AS SPECIFIED IN 2.3.13.21 BUT WILL INHIBIT THE PRINTOUT ONCE THIS LIMIT IS REACHED. AFTER ALL WORDS ARE CHECKED A SUMMARY WILL BE PRINTED:

X WORDS BAD OUT OF 128 WORDS READ

LIMITS 0 - 126

2.3.13.23 MAX HD X?

REPRESENTS MAXIMUM HEAD TO USE IN SEEK OPERATIONS.

LIMITS 0 - 1

2.3.13.26 MIN HD X?

REPRESENTS MINIMUM HEAD TO USE IN SEEK OPERATIONS

LIMITS 0 - 1

2.3.13.27 MAX CYL X?

MAXIMUM INNER CYLINDER TO BE USED IN SEEK OPERATIONS.

LIMITS 0 - 255

2.3.13.28 MIN CYL X?

MINIMUM OUTER CYLINDER TO BE USED IN SEEK OPERATIONS.

LIMITS 0 - 255

2.3.13.29 MAX SEC X?

MAXIMUM SECTOR TO START TRANSFER ON

LIMITS 0 - 39

2.3.13.30 MIN SEC X?

MINIMUM SECTOR TO START TRANSFER ON

LIMITS 0 - 39

AFTER ANSWERING THE LAST SOFTWARE PARAMETER THE PROGRAM WILL START THE TESTING.

2.3.13.31 CHK DRDY X?

ON START UP IF THIS QUESTION IS ANSWERED YES THE PROGRAM WILL NOT TEST ANY DRIVES THAT DO NOT HAVE DRIVE READY HIGH.

LIMITS Y OR N

2.3.14 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

THE FULL CAPABILITY OF THE HARDWARE DIALOGUE IS REVEALED BY THE FOLLOWING DISCUSSION OF WHAT HAPPENS INTERNALLY.

AS SOON AS THE QUESTION "# UNITS?" IS ANSWERED (WITH THE NUMBER N, SAY) SPACE IN CORE IS ALLOCATED FOR N P-TABLES. ALL OF THE P-TABLES ARE OF THE SAME FORMAT, AND THERE IS A ONE-TO ONE CORRESPONDENCE BETWEEN THE HARDWARE PARAMETER QUESTIONS AND THE SLOTS IN THE P-TABLE FORMAT.

ON THE FIRST TRIP THRU THE QUESTIONS, ALL OF THE SLOTS IN ALL OF THE P-TABLES ARE FILLED. IF THE OPERATOR TYPES IN LESS THAN N EXPLICIT VALUES IN RESPONSE TO A PARTICULAR QUESTION, THESE VALUES ARE PLACED IN THE P-TABLES (ONE VALUE GOING INTO THE PROPER SLOT OF EACH P-TABLE BEGINNING WITH THE FIRST P-TABLE) UNTIL THE STRING OF VALUES IS EXHAUSTED. THE LAST VALUE IN THE STRING BECOMES THE NEW DEFAULT AND IS USED THEN AND THERE TO FILL THAT SLOT IN THE REMAINING P-TABLES.

ON SUBSEQUENT TRIPS THRU THE QUESTIONS, THE SAME PROCESS IS CARRIED OUT, EXCEPT THAT THE EARLIEST P-TABLE NOT TO HAVE RECEIVED AN EXPLICIT VALUE IN ANY OF ITS SLOTS NOW ASSUMES THE ROLE THAT TABLE NUMBER ONE PLAYED IN THE FIRST TRIP.

THE SERIES OF QUESTIONS IS REISSUED UNTIL AT LEAST ONE QUESTION HAS RECEIVED N EXPLICIT VALUES FROM THE OPERATOR.

IN GIVING A STRING OF VALUES, COMMAS WITHOUT INTERVENING VALUES MAY BE USED TO INDICATE A REPETITION OF THE LAST NAMED VALUE.

A STRING OF VALUES MAY BE GIVEN AS A RANGE (6-10 FOR EXAMPLE). IF THE VALUES REPRESENT PURE NUMERICAL DATA, THIS SAMPLE RANGE TRANSLATES TO THE STRING 6,7,8,9,10 (AN INCREMENT OF 1). IF THE VALUES ARE ADDRESSES, THE SAMPLE RANGE TRANSLATES TO THE STRING 6,8,10 (AN INCREMENT OF 2).

NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO CONSTRUCT A SET OF P-TABLES. ASSUME THAT WE HAVE 64 UNITS, AND THAT THERE ARE THREE HARDWARE PARAMETERS FOR EACH (THREE SLOTS IN THE P-TABLE, THREE HARDWARE QUESTIONS

IN THE DIALOGUE). LET THE DESIRED VALUE FOR THE FIRST PARAMETER BE THE NUMBER 75 FOR ALL 64 TABLES. LET THE DESIRED VALUE FOR THE SECOND PARAMETER BE EQUAL TO THE UNIT NUMBER (1,2,3,...,64) EXCEPT FOR UNIT 50, WHICH SHOULD RECEIVE THE VALUE 49. LET THE DESIRED VALUE FOR THE THIRD PARAMETER BE THE NUMBER 76 FOR THE FIRST 20 UNITS AND THE NUMBER 77 FOR THE LAST 44 UNITS.

THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:

```
# UNITS (D)  ?  64

UNIT 1
<QUESTION 1> ? 75
<QUESTION 2> ? 1-20
<QUESTION 3> ? 76

UNIT 21
<QUESTION 1> ?
<QUESTION 2> ? 21-49,,51-64
<QUESTION 3> ? 77
```

THE FIRST TIME THE SERIES IS ASKED, SLOT ONE RECEIVES A 75 IN ALL 64 TABLES. SLOT TWO RECEIVES THE VALUES 1,2,3,...,20 IN TABLES 1 THRU 20 AND A CONSTANT 20 IN TABLES 21 THRU 64. SLOT THREE RECEIVES A CONSTANT 76 IN ALL 64 TABLES.

THE SECOND TIME THRU THE SERIES, TABLES 21 THRU THE END ARE GOING TO BE AFFECTED (NOTE THAT THIS PIECE OF INFORMATION IS PRINTED OUT FOR THE THE OPERATOR IN THE FORM "UNIT XX" AT THE BEGINNING OF EACH SERIES). QUESTION 1 IS RESPONDED TO BY A <CR>, SO SLOT ONE STAYS AT CONSTANT 75 IN TABLES 21 THRU 64, SINCE NO NEW EXLICIT VALUES ARE TYPED IN. SLOT TWO GETS THE VALUES 21,22,23,...,49 IN TABLES 21 THRU 49, AND GETS A 49 IN SLOT 50, AND GETS THE VALUES 51,52,53,...,64 IN TABLES 51 THRU 64. SLOT THREE GETS THE VALUE 77 IN TABLES 21 THRU 64.

THE DIALOGUE IS TERMINATED WHEN THE SOFTWARE RECOGNIZES THAT 64 EXPLICIT VALUES HAVE BEEN GIVEN FOR AT LEAST ONE QUESTION (NAMELY QUESTION 2).

2.4 EXECUTION TIMES

EXECUTION TIME IS DEPENDENT ON PROCESSOR, NUMBER OF CONTROLLERS AND DRIVES.

3.0 ERROR INFORMATION

ALL ERRORS ARE PRINTED VIA CONSOLE DEVICE. THE ERROR INCLUDES ERROR NUMBER, TYPE AND PROGRAM LOCATION. ERRORS INCLUDE REGISTERS BEFORE AND AT ERROR WITH RELEVENT DATA.

3.1 ERROR REPORTING

THE FOLLOWING ARE ERROR HEADINGS THAT MAY BE ENCOUNTERED WHILE RUNNING. A BREIF DESCRIPTION IS GIVEN.

```
SFT ERROR

         AN ERROR WAS DISCOVERED, BUT ON RETRY THE ERROR DID NOT
         PERSIST.  INFO GIVEN IS ERROR,RLCS,RLBA, AND RLDA

EXH'D RETRY ON SEEK

         THE NUMBER OF RETRIES GIVEN HAVE FAILED TO POSITION DRIVE
         TO THE GIVEN TRACK.  INFO GIVEN IS RLCS,RLDA,RLBA,LAST POS-
         ITION,PRESENT POSITION, AND DRIVE STATUS

VOL CHK WILL NOT RESET

         A DRIVE RESET WILL NOT RESET VOLUME CHECK BIT

DR DID NOT REC'R FROM PWR UP

         DRIVE DID NOT COME BACK UP AFTER A POWER FAILURE

DATA DMP - DATA CHECK/GARRBLED DATA

         THE PROGRAM ENCOUNTERED A DATA CHECK ERROR BUT WAS
         UNABLE TO MAKE SENSE OUT OF THE FIRST TWO WORDS,
         WHICH ARE USED TO KEY OFF OF.  THEREFORE ALL WORDS OF
         SECTOR ARE DUMPED.(REFER TO SECTION 2.3.13.21)


LIMITS EXCEEDED! HIGH - X   LOW - Y

         ANSWER GIVEN IS NOT WITHIN LIMITS FOR QUESTION.


NO DEFAULT PROVIDED!

         CANNOT <CR> TO THIS QUESTION


ILLEGAL COMMAND

         START, RESTART, CONTINUE, PRINT TYPED IN
         WRONG FORM


ILL ENTRY IN P-TABLE

         ANSWERS IN HARDWARE SECTION ARE NOT LEGAL
                   I.E. MORE THAN TWO CONTROLLERS
                        VECTORS FOR A CONTROLLER NOT CONSISTANT
                        MORE THAN TWO VECTORS.


CAN'T READ FACTORY BAD SECTOR FILE

         PROGRAM IS UNABLE TO READ ANY OF THE FACTORY FILES


CAN'T READ FIELD BAD SECTOR FILE

         PROGRAM IS UNABLE TO READ ANY OF THE FIELD FILES
```

RL01K HAS MORE THAN 16 BAD SECTORS

PROGRAM LIMITS EXERCISING CARTRIDGES TO THOSE WITH LESS THAN 16 BAD SECTORS.

NO DRIVES ENTERED

EITHER NO DRIVES WERE ENTERED OR ALL DRIVES THAT WERE ENTERED WERE DROPPED FOR ONE REASON OR ANOTHER. THE PROGRAM WILL LOOP AFTER PRINTING THE ERROR, WAITING FOR C. A START COMMAND IS NOW NECESSARY.

DRV NOT RDY W/O DRV ERR

ON COMPLETION OF A COMMAND, DRIVE READY IS CHECKED FOR A POSSIBLE DRIFT TRACKING PROBLEM. IF THERE IS NO DRIVE READY A GET STATUS IS DONE TO VERIFY THAT THE DRIVE IS NOT IN PROCESS OF SEEKING. IF IT IS SEEKING THE CONDITION IS LEGAL. THIS TYPEOUT IMPLIES THERE WERE NO DRIVE ERRORS WHICH MAY HAVE CAUSED DRIVE READY TO GO AWAY.

TRCK ERR

THIS ERROR MEANS THAT THE DRIVE IS NO LONGER ON THE TRACK WE WERE ON FOR THE LAST READ HEADER PERFORMED. EACH SEEK IS VERIFIED BY AN IMMEDIATE INITIAL READ HEADER, FROM THAT POINT ANY SUBSEQUENT READ HEADER, READ OR WRITE WILL PRINT THIS ERROR IF THE TRACK IS NOT CORRECT. THIS ERROR WILL PRINT THE POSTION BEFORE THE LAST SEEK, THE PRESENT POSITION AND THE EXPECTED POSITION.

MIS-SK ERR

AFTER A SEEK WAS DONE, READ HEADER IS DONE TO VERIFY THE SEEK. THE ERROR PRINTOUT WILL INCLUDE THE LAST POSTION BEFORE THE SEEK, THE PRESENT POSITION AND THE EXPECTED POSTION.

DRV STAT ERR

THE RESULT OF A GET STATUS OPERATION IS INCORRECT. EITHER A ERROR BIT IS SET OR THE STATE IS WRONG

RE ERR ENC'D

IN ATTEMPTING A RETRY OF A FUNCTION THAT WAS IN ERROR THE RETRY WAS SUCCESSFUL. ERROR INFORMATION CONSISTS OF BUS ADDRESS, DISK ADDRESS, NUMBER OF RETRIES BEFORE SUCCESS AND ERROR TYPE.

HRD ERR

THE NUMBER OF RETRIES WERE EXHAUSTED WITH OUT SUCCESS THE ERROR PRINTOUT CONSISTS OF ALL REGISTERS BEFORE COMMAND AND AT TIME OF ERROR.

INIT WR OF SEC BAD

WHILE WRITING THE PACK INITIALLY THE SECTOR INDICATED COULD NOT BE WRITTEN AND VERIFIED. THIS SECTOR WAS NOT IN THE BAD SECTOR FILE. EITHER STOP THE EXERCISER AND CHANGE CARTRIDGE, STOP THE EXERCISER AND VERIFY THE CARTRIDGE OR IGNORE ALL ERRORS FORM THAT SECTOR.

3.2 ERROR HALTS

ERROR HALTS ARE SUPPORTED PER DESCRIBED IN THE PREVIOUS SECTION WITH /FLAG:HOE. THERE ARE NO OTHER HALTS.

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS

PERFORMANCE REPORTS ARE GIVEN AUTOMATICALLY (PER SOFTWARE PARAMETERS), WHEN A DRIVE IS DROPPED, OR AT OPERATOR REQUEST (PRINT) THE FORMAT IS:

```
          *** RL01 PERFORMANCE REPORT ***

TIME: HH:MM:SS  RLCS: XXXXXX  DRIVE: Y  RUNNING OR DROPPED DH:DM
PACK SERIAL #: NNNNNNNNNN
SEEKS: IIIII
BITS READ:        JJJJJJJJ (*16)
BITS WRITTEN:     KKKKKKKK (*16)

ERRORS
DRIVE:  N         SEEK:   N         TRACK:  N         DATA:   N
HARD:   N         SOFT:   N
```

```
DCK:     N       HCRC:    N       NXM:     N       HNF:     N
DLT:     N       OPI:     N

WHERE:

HH       IS HOURS SINCE START/RESTART
MM       IS MINUTES SINCE START/RESTART
SS       IS SECONDS SINCE START/RESTART
XXXXXX   IS ADDRESS OF CONTROLLER
Y        IS DRIVE NUMBER
DH       IS HOUR AT WHICH DRIVE WAS DROPPED
DM       IS MINUTE AT WHICH DRIVE WAS DROPPED
NNNNNNNNNN - IS 10 DIGIT OCTAL SERIAL NUMBER OF PACK
IIII     IS TOTAL NUMBER OF SEEKS SINCE 0:00:00
JJJJ     IS TOTAL NUMBER OF BITS READ (*16) SINCE 0:00:00
KKKK     IS TOTAL NUMBER OF BITS WRITTEN (*16) SINCE 0:00:00
N        IS NUMBER OF THAT TYPE ERROR SINCE 0:00:00
```

4.2 PROGRESS REPORTS

THE ONLY PROGRESS REPORT IS THE AUTOMATIC PERFORMANCE REPORT

```
5.0 DEVICE INFORMATION TABLES
    -------------------------

        THE RL11/RLV11 CONTROLLER HAS THE FOLLOWING FOUR(4)
        REGISTERS FOR CONTROL OF THE SUBSYSTEM.

        RLCS - CONTROL AND STATUS REGISTER (XXXXX0)
        -------------------------------------------

        BIT 15 - COMPOSITE ERROR
        BIT 14 - DRIVE ERROR
        BIT 13 - NON EXISTANT MEMORY ERROR
        BIT 12 - HEADER NOT FOUND (WITH BIT 10 SET)
               - DATA LATE (WITH BIT 10 CLEAR)
        BIT 11 - HEADER CRC (WITH BIT 10 SET)
               - DATA CRC (WITH BIT 10 CLEAR)
        BIT 10 - OPERATION INCOMPLETE
        BIT  9/8 - DRIVE SELECT (0-3)
        BIT  7 - CONTROLLER READY
        BIT  6 - INTERRUPT ENABLE
        BIT  5 - EXTENDED BUS ADDRESS (BIT 17)
        BIT  4 - EXTENDED BUS ADDRESS (BIT 16)
        BIT 3-1 - FUNCTION CODE
                0 - NOP (PDP-11) MAINT (LSI-11)
                1 - WRITE CHECK
                2 - GET DRIVE STATUS
                3 - SEEK
                4 - READ HEADER
                5 - WRITE DATA
                6 - READ DATA
                7 - READ WITHOUT HEADER COMPARE

        BIT  0 - DRIVE READY


        RLBA - BUS ADDRESS REGISTER (XXXXX2)
        ------------------------------------

        BITS 15-1 BUS ADDRESS OF DATA TRANSFER
        BIT 0 SHOULD BE 0

        RLDA - DISK ADDRESS REGISTER (XXXXX4)
        -------------------------------------

        FOR READ/WRITE FUNCTIONS
        ------------------------

        BIT 15 - MUST BE ZERO(0)
        BIT 14-7 - CYLINDER ADDRESS FOR TRANSFER
        BIT 6 - SURFACE FOR TRANSFER
        BIT 5-0 - SECTOR FOR TRANSFER (0-47)

        FOR SEEK FUNCTION
        -----------------

        BIT 15 - MUST BE ZERO(0)
        BIT 14-7 - DIFFERENCE TO NEW CYLINDER
        BIT 6-5 - MUST BE ZERO(0)
BIT 4 - SURFACE
BIT 3 - MUST BE ZERO
BIT 2 - SEEK DIRECTION( 1 - IN / 0 - OUT )
BIT 1 - MUST BE ZERO
BIT 0 - MUST BE ONE(1)

FOR GET STATUS FUNCTION
-----------------------

BIT 15-4 - IGNORED SHOULD BE ZERO
BIT 3 - DRIVE RESET
BIT 2 - MUST BE ZERO
BIT 1 - MUST BE ONE
BIT 0 - MUST BE ONE

RLMP - MULTIPURPOSE REGISTER
----------------------------

FOR READ/WRITE FUNCTION
-----------------------

BIT 15 - 0 - WORD COUNT(TWO'S COMPLIMENT)

FOR READ HEADER FUNCTION
------------------------

BIT 15-0 - DISK HEADER OF SECTOR (FIRST READ)
         - ZERO WORD (SECOND READ)
         - HEADER CRC (THIRD READ)

FOR GET STATUS FUNCTION
-----------------------

HAS DRIVE STATUS

BIT 15 - WRITE DATA ERROR
BIT 14 - CURRENT HEAD ERROR(CHE)
BIT 13 - WRITE LOCK STATUS(WL)
BIT 12 - SEEK TIME OUT(SKTO)
BIT 11 - SPIN ERROR(SPE)
BIT 10 - WRITE GATE ERROR(WGE)
BIT  9 - VOLUME CHECK(VC)
BIT  8 - DRIVE SELECT ERROR(DSE)
BIT  7 - RESERVED(0)
BIT  6 - SURFACE
BIT  5 - COVER OPEN
BIT  4 - HEADS HOME
BIT  3 - BRUSHES HOME
BIT 2-0 -STATE BITS
        0 - LOAD STATE
        1 - SPIN UP
        2 - BRUSH CYCLE
        3 - LOAD HEADS
        4 - SEEK - TRACK COUNTING
        5 - SEEK - LINEAR MODE
        6 - UNLOAD HEADS
        7 - SPIN DOWN
```

6.0 TEST SUMMARIES

PROGRAM DESCRIPTION

THE PROGRAM WILL TRY TO SIMULATE A USER ENVIRONMENT WITH RANDOM SELECTION OF DRIVES PERFORMING RANDOM OPERATIONS OF GET STATUS, SEEK, READ AND WRITE.

INITIALLY THE BAD SECTOR FILE IS RECOVERED FROM EACH DRIVE AND STORED, THEN EACH PACK IS ENTIRELY WRITTEN RANDOMLY WITH ONE OF EIGHT PREDETERMINED PATTERNS.

THE MAIN LOOP IS A CONTINUOUS LOOP OF THE FOLLOWING STEPS

```
1.      RANDOMLY SELECT A DRIVE

2.      CHECK CONTROLLER OF SELECTED DRIVE IS NOT BUSY;
        THEN STEP 3; ELSE STEP 1

3.      RANDOMLY SELECT FUNCTION FOR DRIVE
         IF WRITE CHECK NEEDED; THEN STEP 4
         IF SEEK NEEDS VERIFICATION; THEN STEP 12
         IF IN PROCESS OF RETRY; THEN STEP 6
         IF IN PROCESS OF SEEK RETRY; THEN STEP 8
         IF GET STATUS; THEN STEP 5
         IF SEEK; THEN STEP 7
         IF READ; THEN STEP 13
         IF WRITE; THEN STEP 17

4.      ISSUE WRITE CHECK; GO TO STEP 1

5.      ISSUE GET STATUS; GO TO STEP 1

6.      ISSUE LAST FUNCTION; GO TO STEP 1

7.      GET RANDOM CYLINDER AND HEAD WITHIN
        SOFTWARE PARAMETER LIMITS

8.      CALCULATE DIFFERENCE TO NEW POSITION

9       ISSUE SEEK

10      SET POSITION VERIFICATION NEEDED FLAG

11      GO TO STEP 1

12.     ISSUE READ HEADER, THEN STEP 1

13.     GET RANDOM WORD COUNT WITHIN LIMITS

14.     GET RANDOM SECTOR WITHIN LIMITS

15.     CHECK THAT WORD COUNT AND SECTOR FIT
        ON TRACK IF THEN STEP 16; ELSE FIX

16.     ISSUE READ; GO TO STEP 1

17.     GET RANDOM WORD COUNT WITHIN LIMITS
```

```
18.      GET RANDOM SECTOR WITHIN LIMITS

19.      CHECK THAT WORD COUNT AND SECTOR FIT
         ON TRACK IF THEN STEP 20; ELSE FIX

20.      SELECT RANDOM PATTERNS IN 128 WORD CHUNKS UNTIL WORD
         COUNT DONE AND WRITE BUFFER IN MEMORY.

21.      ISSUE WRITE; GO TO STEP 1

THE PROGRAM WILL STAY WITHIN THAT MAIN LOOP UNTIL INTERRUPTED
OUT BY A FUNCTION FINISHING AT WHICH TIME THE INTERRUPT
SERVICE ROUTINE WILL START EXECUTION.

1.       READ ALL REGISTERS OF CONTROLLER THAT
         INTERRUPTED AND SAVE IMAGES

2.       IF NO ERROR SET; THEN STEP 3; ELSE STEP 14

3.       CHECK FUNCTION WHICH CAUSED INTERRUPT
         IF WRITE CHECK; THEN STEP 3A
         IF GET STATUS; THEN STEP 5
         IF SEEK; THEN STEP 4A.
         IF READ HEADER; THEN STEP 7
         IF READ; THEN STEP 9
         IF WRITE; THEN STEP 3B

3A.      CLEAR WRITE CHECK NEEDED FLAG, THEN STEP 4

3B.      SET WRITE CHECK NEEDED FLAG IF REQUESTED THEN STEP 4

4.       IF RETRY > 0 THEN REPORT SOFT ERROR, ELSE STEP 4A

4A.      EXIT TO MAIN PROGRAM

5.       CHECK STATUS FOR:        NO ERRORS
                                  COVER CLOSED
                                  BRUSHES HOME
                                  HEADS OUT
                                  SEEK LINEAR/TRACKING

         IF THEN STEP 4; ELSE STEP 6

6.       REPORT STATUS ERROR; GO TO STEP 4A

7.       SET VERIFICATION DONE FLAG COMPARE PRESENT POSITION
         WITH HEADER WORD IF THEN STEP 4A; ELSE STEP 8

8.       REPORT MIS-SEEK, SET NEW POSITION; GO TO STEP 4

9.       IF DATA TO BE COMPARED; THEN STEP 10; ELSE STEP 4

10.      CHECK VALIDITY OF FIRST TWO WORDS: IF THEN STEP 12;
         ELSE STEP 11.

11.      REPORT GARBLED DATA; GO TO STEP 4

12.      CHECK WORDS READ IN IF OKAY THEN STEP 4A
         ELSE STEP 13
```

```
13.     REPORT DATA ERROR, GO TO STEP 4

14.     IF DRIVE ERROR; THEN STEP 33; ELSE STEP 15

15.     IF NXM; THEN STEP 18; ELSE STEP 16

16.     IF OPI; THEN STEP 18; ELSE STEP 17

17.     IF DLT; THEN STEP 18; ELSE STEP 20

18.     IF RETRY < LIMIT THEN STEP 4A, ELSE STEP 19

19.     REPORT HARD ERROR; CLEAR FLAGS; GO TO STEP 4A

20.     IF HCRC; THEN STEP 24; ELSE STEP 21

21.     IF DCRC, THEN STEP 29; ELSE STEP 22

22.     IF HNF, THEN STEP 30; ELSE STEP 23

23.     YOU SHOULD NEVER GET HERE

24.     IF DOING READ/WRITE THEN STEP 25
        IF DOING READ HEADER THEN STEP 26

25.     CHECK IF DA IS BAD SECTOR THEN STEP 4A; ELSE STEP 18.

26.     READ 40 HEADERS, IF ALL GOOD THEN STEP 27; ELSE STEP 28

27.     REPORT SOFT HEADER CRC; GO TO 4A

28.     FIGURE OUT BAD HEADER IF IN FILE THEN STEP 4A; ELSE STEP 18

29.     CHECK IF DA-1 IS IN FILE IF THEN STEP 4A; ELSE STEP 18

30.     READ HEADER. IF ON CORRECT TRACK THEN STEP
        31; ELSE STEP 32

31.     CHECK IF DA IS IN FILE IF THEN STEP 4A, ELSE STEP 18

32.     REPORT TRACKING; FIX POSITION, GO TO STEP 4

33.     ACT UPON:       VC
                        SKTO
                        SPE
                        WGE
                        WDE
                        CHE

34.     GO TO STEP 4
```

7.0 PROGRAM LISTING

```
 2806                                        .ENABLE AMA
 2807                                        .ENABLE ABS
 2808                                        .NLIST  ME,CND,MD
 2809
 2810           002000                       .=2000
 2811
 2812
 2814
 2815   002000                               SVC
 2816           000000                       SVCINS=0
 2817           000000                       SVCTAG=0
 2818
 2819
 2820
 2821   002000                                      POINTER ALL
 2822
 2823
 2824   002000                               BGNMOD  MDHEDR
 2825   002000                                      HEADER  DZRLE,A,0
  (5)   002000     104                              .ASCII  @D@
  (5)   002001     132                              .ASCII  @Z@
  (5)   002002     122                              .ASCII  @R@
  (5)   002003     114                              .ASCII  @L@
  (5)   002004     105                              .ASCII  @E@
  (6)   002005     000                              .BYTE   0
  (6)   002006     000                              .BYTE   0
  (5)   002007     000                              .BYTE   0
  (4)   002010     101                              .ASCII  @A@
  (4)   002011     060                              .ASCII  @0@
  (4)   002012     001                              .BYTE   C$REVISION
  (3)   002013     006                              .BYTE   C$EDIT
  (4)   002014  000000                              .WORD   0
  (4)   002016  000000                              .WORD
  (4)   002020  000000                              .WORD
  (4)   002022  000000                              .WORD
  (4)   002024  000000                              .WORD   0
  (5)   002026  000000                              .WORD   0
  (4)   002030  000000                              .WORD   0
  (4)   002032  000000                              .WORD   0
  (4)   002034  000000                              .WORD   0
  (4)   002036  000000                              .WORD   0
  (4)   002040  007570                              .WORD   L$DISPATCH
  (4)   002042  007650                              .WORD   L$INIT
  (4)   002044  011050                              .WORD   L$CLEAN
  (4)   002046  026176                              .WORD   L$HARD
  (4)   002050  026322                              .WORD   L$SOFT
  (4)   002052  002104                              .WORD   L$DVTYP
  (4)   002054  007572                              .WORD   L$RPT
  (4)   002056  007454                              .WORD   L$HW
  (4)   002060  007470                              .WORD   L$SW
  (4)   002062  002102                              .WORD   L$DR
  (4)   002064  002102                              .WORD   L$DRST
  (4)   002066  000000                              .WORD   0
  (4)   002070  011202                              .WORD   L$AU
  (4)   002072  011266                              .WORD   L$DU
  (4)   002074  000000                              .WORD   0
  (4)  002076  027706                          .WORD   L$LAST
 2826
 2827  002100                                  ENDMOD
 2828
 2829
 2830
 2831  002100                                  DEVREG
  (5)  002100  000000                          .WORD   0
  (2)  002102  000001                          .BLKW
 2832
 2833  002104                                  DEVTYP  <RL01>
  (3)  002104  046122  030460    000           .ASCIZ  @RL01@
  (2)          002112                          .EVEN
 2834
 2835
 2836                                  .SBTTL  BIT AND OFFSET DEFINITIONS
 2837
 2838                                  ;DEFINITIONS
 2839
 2840
 2841  002112                          BGNMOD  GLBEQAT
 2842
 2843  002112                          EQUALS
  (1)                                  ;
  (1)                                  ; BIT DIFINITIONS
  (1)                                  ;
  (1)          100000                  BIT15== 100000
  (1)          040000                  BIT14== 40000
  (1)          020000                  BIT13== 20000
  (1)          010000                  BIT12== 10000
  (1)          004000                  BIT11== 4000
  (1)          002000                  BIT10== 2000
  (1)          001000                  BIT09== 1000
  (1)          000400                  BIT08== 400
  (1)          000200                  BIT07== 200
  (1)          000100                  BIT06== 100
  (1)          000040                  BIT05== 40
  (1)          000020                  BIT04== 20
  (1)          000010                  BIT03== 10
  (1)          000004                  BIT02== 4
  (1)          000002                  BIT01== 2
  (1)          000001                  BIT00== 1
  (1)                                  ;
  (1)          001000                  BIT9==  BIT09
  (1)          000400                  BIT8==  BIT08
  (1)          000200                  BIT7==  BIT07
  (1)          000100                  BIT6==  BIT06
  (1)          000040                  BIT5==  BIT05
  (1)          000020                  BIT4==  BIT04
  (1)          000010                  BIT3==  BIT03
  (1)          000004                  BIT2==  BIT02
  (1)          000002                  BIT1==  BIT01
  (1)          000001                  BIT0==  BIT00
  (1)                                  ;
  (1)                                  ; EVENT FLAG DEFINITIONS
  (1)                                  ;    EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
```

```
(1)                         ;       EF16:EF01 AVAILABLE FOR PROGRAM USE
(1)                         ;
(1)     000040              EF.START==          32.                           ; START COMMAND WAS ISSUED
(1)     000037              EF.RESTART==        31.                           ; RESTART COMMAND WAS ISSUED
(1)     000036              EF.CONTINUE==       30.                           ; CONTINUE COMMAND WAS ISSUED
(1)     000035              EF.NEW==            29.                           ; A NEW PASS HAS BEEN STARTED
(1)     000034              EF.PWR==            28.                           ; A POWER-FAIL/POWER-UP OCCURRED
(1)                         ;
(1)     000020              EF16==  16.
(1)     000017              EF15==  15.
(1)     000016              EF14==  14.
(1)     000015              EF13==  13.
(1)     000014              EF12==  12.
(1)     000013              EF11==  11.
(1)     000012              EF10==  10.
(1)     000011              EF09==  9.
(1)     000010              EF08==  8.
(1)     000007              EF07==  7
(1)     000006              EF06==  6
(1)     000005              EF05==  5
(1)     000004              EF04==  4
(1)     000003              EF03==  3
(1)     000002              EF02==  2
(1)     000001              EF01==  1
(1)                         ;
(1)
(1)                         ; PRIORITY LEVEL DEFINITIONS
(1)                         ;
(1)     000340              PRI07== 340
(1)     000300              PRI06== 300
(1)     000240              PRI05== 240
(1)     000200              PRI04== 200
(1)     000140              PRI03== 140
(1)     000100              PRI02== 100
(1)     000040              PRI01== 40
(1)     000000              PRI00== 0

        000000              CS=0                                    ;CONTROL AND STATUS OFFSET
        000002              BA=2                                    ;BUSADDRESS OFFSET
        000004              DA=4                                    ;DISK ADDRESS OFFSET
        000006              MP=6                                    ;MULTI PURPOSE OFFSET
                            ;CONSTANT OFFSETS FOR INDIVDUAL DRIVE BUFFERS
                            ;THE ONLY POSITION THAT IS CRITICAL IS THAT OF
                            ;""PRPOS""  IT M U S T (MUST) BE THE LAST ENTRY OF THE BUFFER

        000000              SKCNT=0                         ;SEEK OPERATION COUNT
        000002              RXFR1=2                         ;READ OPERATION COUNT (BITS) LOW ORDER
                                                            ; "       "       "     "   HIGH ORDER
        000004              RXFR2=4
        000006              WXFR1=6                         ;WRITE OPERATION COUNT (BITS) LOW ORDER
        000010              WXFR2=10                        ; "       "        "     "    HIGH ORDER
        000012              ERRCNT=12                       ;ERROR COUNT - HARD
        000014              SFTCNT=14                       ;ERROR COUNT - SOFT
        000016              SKECNT=16                       ;SEEK ERROR COUNT
        000020              DERCNT=20                       ;DRIVE ERROR COUNT
        000022              DCRCER=22                       ;DATA CRC ERROR COUNT
        000024              HCRCER=24                       ;HEADER.CRC ERROR COUNT
        000026          DLTCNT=26               ;DATA LATE ERROR COUNT
        000030          OPICNT=30               ;OPERATION INCOMPLETE ERROR COUNT
        000032          HNFERR=32               ;HEADER NOT FOUND ERROR COUNT
        000034          NXMCNT=34               ;NON EXISTANT MEMORY ERROR COUNT
        000036          RETRY=36                ;PRESENT RETRY NUMBER
        000040          BDA=40                  ;    "     DISK ADDRESS CONTENTS
        000042          BMP=42                  ;PRESENT MULTIPURPOSE CONTENTS
        000044          FUNC=44                 ;LAST FUNCTION LOADED
        000046          BCSADR=46               ;CSR IMAGE OF LAST COMMAND
        000050          LSTHDR=50               ;LAST POSITION ON DISK
        000052          RTYPE=52                ;ERROR ON WHICH RECOVERY IS BEING TRIED
        000054          SKCNT1=54               ;LOW SEEK COUNT
        000056          PRFLGS=56               ;INTERNAL FLAGS
        000060          RXFR3=60                ;THIRD ORDER READ COUNT
        000062          WXFR3=62                ;THIRD ORDER WRITE COUNT
        000064          LSTDA=64                ;DISK ADDRESS AT SOFT ERROR
        000066          DIFWD=66                ;LAST DIFFERENCE WORD OF SEEK
        000070          DPHOUR=70               ;HOUR OF DRIVE DROPPED
        000071          DPMIN=71                ;MINUTE OF DRIVE DROPPED
        000072          TRERR=72                ;TRACKING ERRORS COUNT
        000074          DATCER=74               ;DATA CMP ERRORS
        000076          DOWCK=76                ;PERFORM WRITE CHECK
        000100          SERNM1=100              ;SERIAL NUMBER OF CARTRIDGE
        000102          SERNM2=102              ;SERIAL NUMBER OF CARTRIDGE
        000104          DCS=104                 ;CSR ADDRESS
        000106          DRSEL=106               ;DRIVE SELECT BITS(8,9,10)
        000110          BBA=110                 ;PRESENT BUS ADDRESS CONTENTS
        000112          BSECPT=112              ;POINTER TO BAD SECTOR FILE
        000114          RSEEK=114               ;SEEK IN PROCESS OF RECOVERY
        000116          SOFTCS=116              ;CSR OF SOFT ERROR
        000120          PRPOS=120               ;PRESENT POSITION ON DISK

        000001          SKDON=BIT0
        000001          DRDY=BIT0               ;DRIVE READY
        000100          INTEN=BIT6              ;INTERRUPT ENABLE
        100000          ERR=BIT15               ;COMPOSITE ERROR
        040000          DERR=BIT14              ;DRIVE ERROR
        100000          WDE=BIT15               ;WRITE DATA ERROR
        040000          HCE=BIT14               ;HEAD CURRENT ERROR
        020000          WL=BIT13                ;WRITE LOCK
        010000          SKTO=BIT12              ;SEEK TIMEOUT ERROR
        004000          SPE=BIT11               ;SPINDLE TIMEOUT/UNDER/OVER SPEED
        002000          WGE=BIT10               ;WRITE GATE ERROR
        001000          VC=BIT9                 ;VOLUME CHECK
        000400          DSE=BIT8                ;DRIVE SELECT ERROR
        020000          NXM=BIT13               ;NON-EXISTANT MEMORY ERROR
        010000          DLT=BIT12               ;DATA LATE
        004000          DCRC=BIT11              ;DATA CRC ERROR
        004000          HCRC=BIT11              ;HEADER CRC ERROR
        010000          HNF=BIT12               ;HEADER NOT FOUND ERROR
        002000          OPI=BIT10               ;OPERATION INCOMPLETE ERROR
        000200          CRDY=BIT7               ;CONTROLLER READY
        000040          BA17=BIT5               ;EXTENDED BUS ADDRESS BIT 17
        000020          BA16=BIT4               ;EXTENDED BUS ADDRESS BIT 16
        000002          WRCHK=BIT1              ;WITE CHECK FUNCTION CODE
        000004          GSTAT=BIT2              ;GET DRIVE STATUS FUNCTION CODE
        000006                  SEEK=BIT1!BIT2                  ;SEEK FUNCTION CODE
        000010                  RDHDR=BIT3                      ;READ HEADER FUNCTION CODE
        000012                  WRITE=BIT3!BIT1                 ;WRITE FUNCTION CODE
        000014                  READ=BIT3!BIT2                  ;READ FUNCTION CODE
        000013                  DRST=BIT3!BIT1!BIT0             ;DRIVE RESET COMMAND CODE FOR DRIVE COMMAD WORD
        000003                  GSBIT=BIT1!BIT0                 ;GET STATUS COMMAND CODE FOR DRIVE COMMAND WORD
        000001                  MK=BIT0                         ;MARKER BIT FOR DRIVE COMMAND WORD(SEEK,GET STATUS)
        000004                  SIGN=BIT2                       ;DIRECTION FOR SEEK(0=AWAY FROM SPINDLE)
        000020                  SKHS=BIT4                       ;HEAD SELECT FOR SEEK
        000100                  HEAD=BIT6                       ;HEAD SELECT FOR READ,WRITE,GET STATUS

                                ;OFFSET FOR HARDWARE P-TABLE

        000000                  CSR=0
        000002                  VECT=2
        000004                  PRIOR=4
        000006                  DRBT=6
        000010                  CNT=10

                                ;OFFSET FOR SOFTWARE P-TABLE

        000000                  RLT=0
        000002                  ELT=2
        000004                  SET=4
        000006                  DAT=6
        000010                  SKT=10
        000012                  TYT=12
        000014                  RDT=14
        000016                  DDT=16
        000020                  CHFLG=20
        000022                  MXB=22
        000024                  MXH=24
        000026                  MNH=26
        000030                  MXC=30
        000032                  MNC=32
        000034                  MXS=34
        000036                  MNS=36
        000040                  DCKFG=40
        000042                  DRFLG=42
        000044                  MNB=44
        000046                  SEL=46
        000050                  OPFLG=50
        000052                  DET=52
        000054                  ROF=54
        000056                  RAN=56
        000060                  PAT=60
        000062                  SRLT=62
        000064                  CLMT=64
        000066                  AUTO=66
        000070                  STIP=70
        000072                  WCK=72
        000074                  DCD=74


  002112                                ENDMOD
```

```
                                             ;

                                             .SBTTL  GLOBAL DATA AND CONSTANTS

        002112                               BGNMOD  GLBDAT

        002112  000000                       RECNT:   .WORD   0                        ;READ ERROR COUNT
        002114  000000                       RWCNT:   .WORD   0                        ;R/W ERROR COUNT
        002116  000000                       WHY:     .WORD   0                        ;REASON FOR DROPPING DRIVE
        002120     000                       DRUT:    .BYTE   0                        ;DRIVES UNDER TEST
        002121     000                       DRPRS:   .BYTE   0                        ;DRIVES PRESENT
        002122  000000                       SYSMSK:  .WORD   0                        ;MASK FOR 0-7 DRIVES
        002124  176543                       HINUM:   .WORD   176543                   ;PRIME FOR RANDOM
        002126  123456                       LONUM:   .WORD   123456                   ;NUMBER GENERATOR
        002130  100177                       CYLMSK:  .WORD   100177                   ;MASK FOR CYLINDER ONLY
        002132  100077                       SECMSK:  .WORD   100077                   ;MASK OUT SECTOR BITS
                                             ;
                                             ;THE FOLLOWING LOCATIONS ARE CLEARED AS A GROUP (DOWN TO 'STFLG')
                                             ;THEREFORE DON'T INSERT ANY CONSTANTS
                                             ;
        002134  174400                       CNTLR1:  .WORD   174400                   ;CSR OF CONTROLLER 1 (LUN 0-3)
        002136  000000                       CNTLR2:  .WORD   0                        ;CSR OF CONTROLLER 2 (LUN 4-7)
        002140  000000                       LSTDR1:  .WORD   0                        ;BUFFER POINTER OF DRIVE
        002142  000000                       LSTDR2:  .WORD   0                        ;BUFFER POINTER OF DRIVE
        002144  000000                       BCSR:    .WORD   0                        ;CSR FROM P-TABLE
        002146  000000                       BVEC:    .WORD   0                        ;VECTOR "    "
        002150  000000                       BPRIOR:  .WORD   0
        002152  000000                       BDRSEL:  .WORD   0                        ;DRIVE  "    "
        002154  000000                       HDRFND:  .WORD   0                        ;FLAG TO INDICATE HDR IN BAD LIST
        002156  000000                       CHKSEC:  .WORD   0                        ;SECTOR OF ERROR - USED BY BAD SECTOR LOCATION
        002160  000000                       DECNT:   .WORD   0                        ;DATA ERROR COUNT
        002162  000000                       TEMP0:   .WORD   0                        ;TEMP LOCATION
        002164  000000                       TEMP1:   .WORD   0                        ;TEMP LOCATION
        002166  000000                       TEMP2:   .WORD   0                        ;TEMP LOCATION
        002170  000000                       TEMP3:   .WORD   0                        ;  "     "
        002172  000000                       TEMP4:   .WORD   0                        ;  "     "
        002174  000000                       TEMP5:   .WORD   0                        ;  "     "
        002176  000000                       TEMP6:   .WORD   0                        ;  "     "
        002200  000000                       TEMP7:   .WORD   0                        ;  "     "
        002202  000000                       TEMP8:   .WORD   0                        ;  "     "
        002204  000000                       TEMP9:   .WORD   0                        ;  "     "
        002206  000330                       VECT1:   .WORD   330                      ;VECTOR OF FIRST CONTROLLER
        002210  000000                       VECT2:   .WORD   0                        ;VECTOR  "  2ND
        002212  000000                       PRIOR1:  .WORD   0
        002214  000000                       PRIOR2:  .WORD   0
        002216  000000                       GDDAT:   .WORD   0
        002220  000000                       RNTEMP:  .WORD   0                        ;     "    "
        002222  000000                       INTERVAL:          .WORD   0              ;TIME BETWEEN REPORTS
        002224  000000                       LSTTIM:  .WORD   0                        ;LAST TIME ON SYSTEM CLOCK
        002226  000000                       SECOND:  .WORD   0                        ;SECONDS OF SYSTEM CLOCK
        002230  000000                       MINUTE:  .WORD   0                        ;MINUTES OF SYSTEM CLODK
        002232  000000                       HOUR:    .WORD   0                        ;HOURS OF SYSTEM CLOCK
        002234  000000                       E.CS:    .WORD   0                        ;IMAGES OF REGISTERS
        002236  000000                       E.BA:    .WORD   0                        ;ON INTERRUPT
        002240  000000                       E.DA:    .WORD   0                        ;
        002242  000000                       E.MP:    .WORD   0
```

```
 3032  002244  000000                          E.MP1:   .WORD   0                  ;
 3033  002246  000000                          E.MP2:   .WORD   0                  ;
 3034  002250  000000                          SYSCLK:  .WORD   0                  ;FLAG INDICATING PRESENSE OF SYSTEM CLOCK
 3035  002252  000000                          BUF1:    .WORD   0                  ;BUFFER FOR  FIRST CONTROLLER
 3036  002254  000000                          BUF2:    .WORD   0                  ;BUFFER FOR SECOND CONTROLLER
 3037  002256  000000                          MAXWC:   .WORD   0                  ;MAX WORD COUNT DETERMINED BY CORE
 3038  002260  000000                          UUT:     .WORD   0                  ;NUMBER OF UNITS ON SYSTEM
 3039  002262  000000                          PWRFLG:  .WORD   0                  ;POWER FAIL  INDICATOR
 3040  002264  000000                          TRPFLG:  .WORD   0                  ;TRAP OCCURANCE FLAG
 3041  002266  000000                          STFLG:   .WORD   0                  ;START FLAG
 3042                                          ;
 3043                                          ;END OF MASS CLEAR
 3044                                          ;
 3045  002270  000000                          CNTFLG:  .WORD   0                  ;CONTINUE FLAG
 3046  002272  000000                          FASCII:  .WORD   0                  ;ASCII MESSAGE OF FUNCTION
 3047  002274  000000                          FASPNT:  .WORD   0                  ;POINTER
 3048  002276  000000                          DWCNT:   .WORD   0                  ;ERROR COUNT
 3049  002300  000000                          DWCNT1:  .WORD   0                  ;ERROR COUNT
 3050  002302  000004                          ERRVEC:  .WORD   4                  ;ERROR VECTOR
 3051  002304  000034                          ST1:     .WORD   34                 ;STATES ALLOWED
 3052  002306  000035                          ST2:     .WORD   35                 ;STATES ALLOWED
 3053  002310  000000                          OPCALL:  .WORD   0
 3054  002312  000000                          INCALL:  .WORD   0
 3055
 3056  002314                                           ENDMOD
 3057
 3058
 3059                                           .SBTTL  GLOBAL MESSAGES
 3060
 3061  002314                                  BGNMOD  GLBTXT
 3062
 3063                                          ;GLOBAL TEXT
 3064
 3065
 3069
 3070  002314  044524  042515  020072  TIME:    .ASCIZ  "TIME:  "
 3071  002323     040  046122  051503  MRLCS:   .ASCIZ  " RLCS:  "
 3072  002333     040  051050  041514  CRLCS:   .ASCIZ  " (RLCS):  "
 3073  002345     040  052506  041516  MFUNC:   .ASCIZ  " FUNCTION:  "
 3074  002361     040  051050  041114  CRLBA:   .ASCIZ  " (RLBA)   "
 3075  002373     040  051050  042114  CRLDA:   .ASCIZ  " (RLDA):  "
 3076  002405     040  051050  046514  CRLMP:   .ASCIZ  " (RLMP):  "
 3077
 3078  002417     104  043111  053440  DIFMSG:  .ASCIZ  /DIF WD:  /
 3079  002430  040520  045503  051440  CART:    .ASCIZ  /PACK SERIAL #:  /
 3080  002450  047516  041440  042122  NOCRDY:  .ASCIZ  /NO CRDY/
 3081  002460  051104  047040  052117  DNRDY:   .ASCIZ  /DR NOT RDY/
 3082  002473     104  020122  047516  NORDY:   .ASCIZ  %DR NOT RDY W/O DR ERR%
 3083  002521     102  043525     000  PRGER:   .ASCIZ  /BUG/
 3084  002525     111  044516  020124  NWRTS:   .ASCIZ  /INIT WR OF SEC BAD/
 3085  002550  051440  041505  047524  SMSG:    .ASCIZ  / SECTOR:  /
 3086  002562  047516  043440  047517  EXHAUS:  .ASCIZ  /NO GOOD HDR/
 3087  002576  047125  044504  043501  UDERR:   .ASCIZ  /UNDIAGNOSABLE ERR/
 3088  002620  042523  045505  042440  MSKER:   .ASCIZ  /SEEK ERR/
 3089  002631     123  043117  020124  MSFER:   .ASCIZ  /SOFT ERR ENC'D/
 3090  002650  051104  042440  051122  DRVER:   .ASC'Z  /DR ERR/
```

```
002657     104  020122  051105  MDERS:   .ASCIZ  /DR ERR WILL NOT RESET/
002705     104  020122  052123  MDSER:   .ASCIZ  /DR STAT ERR/
002721     126  046117  041440  MVCER:   .ASCIZ  /VOL CHK WILL NOT CLR/
002746  051127  043440  052101  WGEST:   .ASCIZ  /WR GATE ERR WILL NOT RESET/
003001     104  020122  051105  MRDER:   .ASCIZ  /DR ERR - RECOVERED/
003024  040504  040524  041440  MDCER:   .ASCIZ  /DATA CMP ERR/
003041     110  051101  020104  MHDER:   .ASCIZ  /HARD ERROR/
003054  040504  040524  042040  DMPDCK:  .ASCIZ  /DATA DUMP - DCK/
003074  051124  041501  044513  TRACK:   .ASCIZ  /TRACKING ERR/
003111     110  042122  042440  ERLMTM:  .ASCIZ  /HRD ERR LMT EXC'D/
003133     123  020113  051105  SERLMT:  .ASCIZ  /SK ERR LMT EXC'D/
003154  043123  020124  051105  SFEMSG:  .ASCIZ  /SFT ERR LMT EXC'D/
003176  040504  040524  042440  DCDMSG:  .ASCIZ  /DATA ERR LMT EXC'D/
003221     104  020122  051105  DERMSG:  .ASCIZ  /DR ERR LMT EXC'D/
003242  052502  043106  051105  OVER:    .ASCIZ  /BUFFER CHOSEN TOO BIG - WAS /
003277     122  050505  041040  REQ:     .ASCIZ  /REQ BY OPR/
003312  054105  023510  020104  SEXHAU:  .ASCIZ  /EXH'D RETRY ON SEEK/
003336  042110  020123  047516  UNLOAD:  .ASCIZ  /HDS NOT UNLD ON ERR/
003362  051104  053440  042114  NOLOAD:  .ASCIZ  /DR WLD NOT LD/
003400  050117  051105  046040  SOPLMT:  .ASCIZ  /OPER LMTS EXC'D/
003420  040507  041122  046102  NOREV:   .ASCIZ  /GARBBLED DATA - CAN'T CHECK IT/
003457     115  051117  020105  MBDMSC:  .ASCIZ  /MORE THAN 16 BAD SECTORS/
003510  047516  043040  041501  HWSEC:   .ASCIZ  /NO FACTORY FILE/
003530  047516  043040  042511  SWSEC:   .ASCIZ  /NO FIELD FILE/
003546  026520  040524  046102  MPT:     .ASCIZ  /P-TABLE: /
003560  046111  020114  026520  ILLEG:   .ASCIZ  /ILL P-TABLE/
003574  053040  041505  047524  MVEC:    .ASCIZ  / VECTOR: /
003606  047516  042040  044522  NODRIV:  .ASCIZ  /NO DRIVES/
003620  042040  044522  042526  DRNM:    .ASCIZ  / DRIVE: /
003631     040  051514  020124  LPS:     .ASCIZ  / LST POS: /
003644  042440  050130  050040  EPS:     .ASCIZ  / EXP POS: /
003657     040  042522  020103  RPS:     .ASCIZ  / REC POS: /
003672  051104  042040  042111  NOPWR:   .ASCIZ  /DR DID REC'R FROM PWR UP/
003723     101  020124  052502  BUSAD:   .ASCIZ  /AT BUS ADDR: /
003741     122  052105  054522  MRT:     .ASCIZ /RETRYS: /
003752  042440  051122  051117  ERT:     .ASCIZ  / ERROR TYPE: /
003770  052123  052101  051525  MST:     .ASCIZ  /STATUS WAS: /
004005     040  044123  052517  MST1:    .ASCIZ  / SHOULD BE: /
004022  051040  052105  044522  RT1:     .ASCIZ  / RETRIES ATTEMPTED/
004045     040  054105  023520  EXP:     .ASCIZ  / EXP'D: /
004056  051040  041505  042047  RCD:     .ASCIZ  / REC'D: /
004067     104  044522  042526  DROP:    .ASCIZ  /DRIVE DROPPED/
004105     040  047110  000106  MTHNF:   .ASCIZ  / HNF/
004112  044040  051103  000103  MTHCRC:  .ASCIZ  / HCRC/
004120  042040  045503     000  MTDCRC:  .ASCIZ  / DCK/
004125     040  046104  000124  MTDLT:   .ASCIZ  / DLT/
004132  047440  044520     000  MTOPI:   .ASCIZ  / OPI/
004137     040  054116  000115  MTNXM:   .ASCIZ  / NXM/
004144  042040  053122     000  MTDRV:   .ASCIZ  / DRV/
004151     124  051505  044524  MSTART:  .ASCIZ  /TESTING STARTED/
004171     127  044522  044524  MSWRPK:  .ASCIZ  /WRITING PACK  /
                                ;
                                ;THIS LIST OF ASCII TEXT IS USED AS A TABLE FOR PRINTING
                                ;FUNCTIONS IN ERROR MESSAGES   TABLE IS "MTCR - MTRD".
                                ;THE ORDER IS IMPORTANT AS WELL AS THE LENGTH OF EACH
                                ;ASCIZ STRING. EACH STRING IS SEVEN(10) BYTES PLUS ZERO
```

```
 3147                                     ;FILL BYTE (TOTAL 8(EIGHT) BYTES) LONG.  USED IN LINE1
 3148                                     ;SUBROUTINE..........................................
 3149                                     ;....................................................
 3150
 3151  004210  053440  041522  045510  MTCR:     .ASCIZ  / WRCHK /
 3152  004220  043440  051524  040524  MTGS:     .ASCIZ  / GTSTAT/
 3153  004230  051440  042505  020113  MTSK:     .ASCIZ  / SEEK  /
 3154  004240  051040  044104  051104  MTRH:     .ASCIZ  / RDHDR /
 3155  004250  053440  044522  042524  MTWR:     .ASCIZ  / WRITE /
 3156  004260  051040  040505  020104  MTRD:     .ASCIZ  / READ  /
 3157
 3158                                     ;....................................................
 3159                                     ;END OF LIST NOW YOU CAN PUT ANY THING YOU WANT HERE
 3160                                     ;....................................................
 3161
 3162
 3163
 3167
 3168                                     .EVEN
 3169
 3170  004270                                       ENDMOD
 3171
 3172                                     .SBTTL  ERROR MESSAGES
 3173
 3174  004270                             BGNMOD  GLBERR
 3175
 3176                                                                         ;GENERAL ERROR REPORT
 3177
 3178  004270                             BGNMSG  ERR1
 3179  004270  004737  005460                     JSR     PC,LINE3
 3180  004274                                     ENDMSG
  (3)  004274                             L10000:
  (3)  004274  104023                             EMT     C$MSG
 3181
 3182                                                                         ;MIS-SEEK ERROR REPORT
 3183
 3184  004276                             BGNMSG  ERR2
 3185  004276  004737  005460                     JSR     PC,LINE3
 3186  004302                                     PRINTB  #FMT4,#DIFMSG,DIFWD(R4),#LPS,LSTHDR(R4),#EPS,PRPOS(R4),#RPS,R1
 (15)  004302  010146                             MOV     R1,-(SP)
 (14)  004304  012746  003657                     MOV     #RPS,-(SP)
 (13)  004310  016446  000120                     MOV     PRPOS(R4),-(SP)
 (12)  004314  012746  003644                     MOV     #EPS,-(SP)
 (11)  004320  016446  000050                     MOV     LSTHDR(R4),-(SP)
 (10)  004324  012746  003631                     MOV     #LPS,-(SP)
  (9)  004330  016446  000066                     MOV     DIFWD(R4),-(SP)
  (8)  004334  012746  002417                     MOV     #DIFMSG,-(SP)
  (7)  004340  012746  005777                     MOV     #FMT4,-(SP)
  (6)  004344  012746  000011                     MOV     #11,-(SP)
  (3)  004350  010600                             MOV     SP,R0
  (4)  004352  104014                             EMT     C$PNTB
  (4)  004354  062706  000024                     ADD     #24,SP
 3187  004360                                     ENDMSG
  (3)  004360                             L10001:
  (3)  004360  104023                             EMT     C$MSG
 3188
 3189                                                                           ;SOFT.ERROR RECOVERABLE ERROR REPORT
 3190   004362                                  BGNMSG  ERR3
 3191   004362  004737  005214                          JSR     PC,LINE1
 3192   004366                                          PRINTB  #FMT2A,#CRLCS,SOFTCS(R4),#CRLBA,@BBA(R4),#CRLDA,LSTDA(R4)
  (13)  004366  016446  000064                          MOV     LSTDA(R4),-(SP)
  (12)  004372  012746  002373                          MOV     #CRLDA,-(SP)
  (11)  004376  017446  000110                          MOV     @BBA(R4),-(SP)
  (10)  004402  012746  002361                          MOV     #CRLBA,-(SP)
  (9)   004406  016446  000116                          MOV     SOFTCS(R4),-(SP)
  (8)   004412  012746  002333                          MOV     #CRLCS,-(SP)
  (7)   004416  012746  005712                          MOV     #FMT2A,-(SP)
  (6)   004422  012746  000007                          MOV     #7,-(SP)
  (3)   004426  010600                                  MOV     SP,R0
  (4)   004430  104014                                  EMT     C$PNTB
  (4)   004432  062706  000020                          ADD     #20,SP
 3193   004436                                          PRINTB  #FMT5,#MRT,RETRY(R4),#ERT,RTYPE(R4)
  (11)  004436  016446  000052                          MOV     RTYPE(R4),-(SP)
  (10)  004442  012746  003752                          MOV     #ERT,-(SP)
  (9)   004446  016446  000036                          MOV     RETRY(R4),-(SP)
  (8)   004452  012746  003741                          MOV     #MRT,-(SP)
  (7)   004456  012746  006030                          MOV     #FMT5,-(SP)
  (6)   004462  012746  000005                          MOV     #5,-(SP)
  (3)   004466  010600                                  MOV     SP,R0
  (4)   004470  104014                                  EMT     C$PNTB
  (4)   004472  062706  000014                          ADD     #14,SP
 3194   004476                                          ENDMSG
  (3)   004476                                  L10002:
  (3)   004476  104023                                  EMT     C$MSG
 3195
 3196                                                                           ;GET STATUS ERROR REPORT
 3197
 3198   004500                                  BGNMSG  ERR4
 3199   004500  004737  005460                          JSR     PC,LINE3
 3200   004504                                          PRINTB  #FMT6,#MST,E.MP,#MST1,ST1,ST2
  (12)  004504  013746  002306                          MOV     ST2,-(SP)
  (11)  004510  013746  002304                          MOV     ST1,-(SP)
  (10)  004514  012746  004005                          MOV     #MST1,-(SP)
  (9)   004520  013746  002242                          MOV     E.MP,-(SP)
  (8)   004524  012746  003770                          MOV     #MST,-(SP)
  (7)   004530  012746  006044                          MOV     #FMT6,-(SP)
  (6)   004534  012746  000006                          MOV     #6,-(SP)
  (3)   004540  010600                                  MOV     SP,R0
  (4)   004542  104014                                  EMT     C$PNTB
  (4)   004544  062706  000016                          ADD     #16,SP
 3201   004550                                          ENDMSG
  (3)   004550                                  L10003:
  (3)   004550  104023                                  EMT     C$MSG
 3202
 3203
 3204                                                                           ;DATA ERROR SUMMARY
 3205
 3206   004552                                  BGNMSG  ERR6
 3207   004552  004737  005374                          JSR     PC,LINE2
 3208   004556  016400  000042                          MOV     BMP(R4),R0
 3209   004562                                          PRINTB  #FMT9A,DECNT,R0
  (9)   004562  010046                                  MOV     R0,-(SP)
```

```
    (8)  004564  013746  002160                    MOV     DECNT,-(SP)
    (7)  004570  012746  006214                    MOV     #FMT9A,-(SP)
    (6)  004574  012746  000003                    MOV     #3,-(SP)
    (3)  004600  010600                            MOV     SP,R0
    (4)  004602  104014                            EMT     C$PNTB
    (4)  004604  062706  000010                    ADD     #10,SP
   3210  004610                                    ENDMSG
    (3)  004610                           L10004:
    (3)  004610  104023                            EMT     C$MSG
   3211
   3212                                                                        ;NON RECOVERABLE ERROR REPORT
   3213
   3214  004612                           BGNMSG   ERR7
   3215  004612                                    PRINTB  #FMT8,RETRY(R4),#RT1
    (9)  004612  012746  004022                    MOV     #RT1,-(SP)
    (8)  004616  016446  000036                    MOV     RETRY(R4),-(SP)
    (7)  004622  012746  006146                    MOV     #FMT8,-(SP)
    (6)  004626  012746  000003                    MOV     #3,-(SP)
    (3)  004632  010600                            MOV     SP,R0
    (4)  004634  104014                            EMT     C$PNTB
    (4)  004636  062706  000010                    ADD     #10,SP
   3216  004642  004737  005460                    JSR     PC,LINE3
   3217  004646                                    ENDMSG
    (3)  004646                           L10005:
    (3)  004646  104023                            EMT     C$MSG
   3218
   3219                                                                        ;BAD DATA COMPARE ERROR REPORT
   3220
   3221  004650                           BGNMSG   ERR8
   3222  004650                                    PRINTB  #FMT10,#TIME,HOUR,MINUTE,SECOND,#MRLCS,DCS(R4),#DRNM,<B,DRSEL+1(R4)>
   (15)  004650  005046                            CLR     -(SP)
   (15)  004652  156416  000107                    BISB    DRSEL+1(R4),(SP)
   (14)  004656  012746  003620                    MOV     #DRNM,-(SP)
   (13)  004662  016446  000104                    MOV     DCS(R4),-(SP)
   (12)  004666  012746  002323                    MOV     #MRLCS,-(SP)
   (11)  004672  013746  002226                    MOV     SECOND,-(SP)
   (10)  004676  013746  002230                    MOV     MINUTE,-(SP)
    (9)  004702  013746  002232                    MOV     HOUR,-(SP)
    (8)  004706  012746  002314                    MOV     #TIME,-(SP)
    (7)  004712  012746  006266                    MOV     #FMT10,-(SP)
    (6)  004716  012746  000011                    MOV     #11,-(SP)
    (3)  004722  010600                            MOV     SP,R0
    (4)  004724  104014                            EMT     C$PNTB
    (4)  004726  062706  000024                    ADD     #24,SP
   3223  004732                                    PRINTB  #FMT10A,#CRLBA,@BBA(R4),#CRLDA,BDA(R4),#EXP,GDDAT,#RCD,(R2)
   (15)  004732  011246                            MOV     (R2),-(SP)
   (14)  004734  012746  004056                    MOV     #RCD,-(SP)
   (13)  004740  013746  002216                    MOV     GDDAT,-(SP)
   (12)  004744  012746  004045                    MOV     #EXP,-(SP)
   (11)  004750  016446  000040                    MOV     BDA(R4),-(SP)
   (10)  004754  012746  002373                    MOV     #CRLDA,-(SP)
    (9)  004760  017446  000110                    MOV     @BBA(R4),-(SP)
    (8)  004764  012746  002361                    MOV     #CRLBA,-(SP)
    (7)  004770  012746  006322                    MOV     #FMT10A,-(SP)
    (6)  004774  012746  000011                    MOV     #11,-(SP)
    (3)  005000  010600                            MOV     SP,R0
   (4)  005002  104014                         EMT     C$PNTB
   (4)  005004  062706  000024                 ADD     #24,SP
  3224  005010                                 PRINTB  #FMT10B,R2
   (8)  005010  010246                         MOV     R2,-(SP)
   (7)  005012  012746  006373                 MOV     #FMT10B,-(SP)
   (6)  005016  012746  000002                 MOV     #2,-(SP)
   (3)  005022  010600                         MOV     SP,R0
   (4)  005024  104014                         EMT     C$PNTB
   (4)  005026  062706  000006                 ADD     #6,SP
  3225  005032                                 ENDMSG
   (3)  005032                         L10006:
   (3)  005032  104023                         EMT     C$MSG
  3226                                                                                  ;DRIVE ERROR
  3227
  3228  005034                         BGNMSG  ERR9
  3229
  3230  005034  004737  005460                 JSR     PC,LINE3
  3231  005040                                 PRINTB  #FMT13,#MST,R1,#LPS,LSTHDR(R4)
  (11)  005040  016446  000050                 MOV     LSTHDR(R4),-(SP)
  (10)  005044  012746  003631                 MOV     #LPS,-(SP)
   (9)  005050  010146                         MOV     R1,-(SP)
   (8)  005052  012746  003770                 MOV     #MST,-(SP)
   (7)  005056  012746  006431                 MOV     #FMT13,-(SP)
   (6)  005062  012746  000005                 MOV     #5.-(SP)
   (3)  005066  010600                         MOV     SP,P0
   (4)  005070  104014                         EMT     C$PNTB
   (4)  005072  062706  000014                 ADD     #14,SP
  3232  005076                                 ENDMSG
   (3)  005076                         L10007:
   (3)  005076  104023                         EMT     C$MSG
  3233
  3234
  3235                                                                                  ;INVALID ENTRY IN P-TABLE REPORT
  3236
  3237  005100                         BGNMSG  ERR10
  3238  005100                                 PRINTB  #FMT11,#MPT,R1,#MRLCS,BCSR,#MVEC,BVEC
  (13)  005100  013746  002146                 MOV     BVEC,-(SP)
  (12)  005104  012746  003574                 MOV     #MVEC,-(SP)
  (11)  005110  013746  002144                 MOV     BCSR,-(SP)
  (10)  005114  012746  002323                 MOV     #MRLCS,-(SP)
   (9)  005120  010146                         MOV     R1,-(SP)
   (8)  005122  012746  003546                 MOV     #MPT,-(SP)
   (7)  005126  012746  006401                 MOV     #FMT11,-(SP)
   (6)  005132  012746  000007                 MOV     #7,-(SP)
   (3)  005136  010600                         MOV     SP,R0
   (4)  005140  104014                         EMT     C$PNTB
   (4)  005142  062706  000020                 ADD     #20,SP
  3239  005146                                 ENDMSG
   (3)  005146                         L10010:
   (3)  005146  104023                         EMT     C$MSG
  3240
  3241
  3242  005150                         BGNMSG  ERR12
  3243
  3244  005150  004737  005460                 JSR     PC,LINE3
  3245
```

```
3246  005154                                          ENDMSG
 (3)  005154                                  L10011:
 (3)  005154  104023                                  EMT     C$MSG
3247
3248  005156                                  BGNMSG  ERR13
3249
3250  005156  004737  005460                          JSR     PC,LINE3
3251  005162                                          PRINTB  #FMT12,#SMSG,BDA(R4)
 (9)  005162  016446  000040                          MOV     BDA(R4),-(SP)
 (8)  005166  012746  002550                          MOV     #SMSG,-(SP)
 (7)  005172  012746  006421                          MOV     #FMT12,-(SP)
 (6)  005176  012746  000003                          MOV     #3,-(SP)
 (3)  005202  010600                                  MOV     SP,R0
 (4)  005204  104014                                  EMT     C$PNTB
 (4)  005206  062706  000010                          ADD     #10,SP
3252
3253  005212                                          ENDMSG
 (3)  005212                                  L10012:
 (3)  005212  104023                                  EMT     C$MSG
3254
3255  005214  016437  000044  002274          LINE1:  MOV     FUNC(R4),FASPNT         ;GET FUNCTION
3256  005222  012737  004210  002272                  MOV     #MTCR,FASCII            ;FIRST FUNCTION ASCIZ
3257  005230  042737  000100  002274                  BIC     #INTEN,FASPNT           ;CLEAR INTERRUPT ENABLE
3258  005236  006237  002274                          ASR     FASPNT                  ;ALIGN
3259  005242  005337  002274                  1$:     DEC     FASPNT                  ;DOWN COUNT FUNCTION
3260  005246  001404                                  BEQ     2$                      ;FOUND?
3261  005250  062737  000010  002272                  ADD     #8.,FASCII              ;NO NEXT ONE
3262  005256  000771                                  BR      1$                      ;LOOP
3263
3264  005260                                  2$:
3265
3266  005260                                          PRINTB  #FMT1,#TIME,HOUR,MINUTE,SECOND,#MRLCS,DCS(R4),#DRNM,<B,DRSEL+1(R4)>
(15)  005260  005046                                  CLR     -(SP)
(15)  005262  156416  000107                          BISB    DRSEL+1(R4),(SP)
(14)  005266  012746  003620                          MOV     #DRNM,-(SP)
(13)  005272  016446  000104                          MOV     DCS(R4),-(SP)
(12)  005276  012746  002323                          MOV     #MRLCS,-(SP)
(11)  005302  013746  002226                          MOV     SECOND,-(SP)
(10)  005306  013746  002230                          MOV     MINUTE,-(SP)
 (9)  005312  013746  002232                          MOV     HOUR,-(SP)
 (8)  005316  012746  002314                          MOV     #TIME,-(SP)
 (7)  005322  012746  005626                          MOV     #FMT1,-(SP)
 (6)  005326  012746  000011                          MOV     #11,-(SP)
 (3)  005332  010600                                  MOV     SP,R0
 (4)  005334  104014                                  EMT     C$PNTB
 (4)  005336  062706  000024                          ADD     #24,SP
3267  005342                                          PRINTB  #FMT1A,#MFUNC,FASCII
 (9)  005342  013746  002272                          MOV     FASCII,-(SP)
 (8)  005346  012746  002345                          MOV     #MFUNC,-(SP)
 (7)  005352  012746  005662                          MOV     #FMT1A,-(SP)
 (6)  005356  012746  000003                          MOV     #3,-(SP)
 (3)  005362  010600                                  MOV     SP,R0
 (4)  005364  104014                                  EMT     C$PNTB
 (4)  005366  062706  000010                          ADD     #10,SP
3268  005372  000207                                  RTS     PC
3269
```

```
 3270  005374                            LINE2:   PRINTB  #FMT9,#TIME,HOUR,MINUTE,SECOND,#MRLCS,DCS(R4),#DRNM,<B,DRSEL+1(R4)>
  (15) 005374  005046                             CLR     -(SP)
  (15) 005376  156416  000107                     BISB    DRSEL+1(R4),(SP)
  (14) 005402  012746  003620                     MOV     #DRNM,-(SP)
  (13) 005406  016446  000104                     MOV     DCS(R4),-(SP)
  (12) 005412  012746  002323                     MOV     #MRLCS,-(SP)
  (11) 005416  013746  002226                     MOV     SECOND,-(SP)
  (10) 005422  013746  002230                     MOV     MINUTE,-(SP)
  (9)  005426  013746  002232                     MOV     HOUR,-(SP)
  (8)  005432  012746  002314                     MOV     #TIME,-(SP)
  (7)  005436  012746  006156                     MOV     #FMT9,-(SP)
  (6)  005442  012746  000011                     MOV     #11,-(SP)
  (3)  005446  010600                             MOV     SP,R0
  (4)  005450  104014                             EMT     C$PNTB
  (4)  005452  062706  000024                     ADD     #24,SP
 3271  005456  000207                             RTS     PC
 3272
 3273  005460  004737  005214            LINE3:   JSR     PC,LINE1
 3274  005464                                     PRINTB  #FMT2,#CRLCS,BCSADR(R4),#CRLBA,@BBA(R4),#CRLDA,BDA(R4),#CRLMP,BMP(R4)
  (15) 005464  016446  000042                     MOV     BMP(R4),-(SP)
  (14) 005470  012746  002405                     MOV     #CRLMP,-(SP)
  (13) 005474  016446  000040                     MOV     BDA(R4),-(SP)
  (12) 005500  012746  002373                     MOV     #CRLDA,-(SP)
  (11) 005504  017446  000110                     MOV     @BBA(R4),-(SP)
  (10) 005510  012746  002361                     MOV     #CRLBA,-(SP)
  (9)  005514  016446  000046                     MOV     BCSADR(R4),-(SP)
  (8)  005520  012746  002333                     MOV     #CRLCS,-(SP)
  (7)  005524  012746  005671                     MOV     #FMT2,-(SP)
  (6)  005530  012746  000011                     MOV     #11,-(SP)
  (3)  005534  010600                             MOV     SP,R0
  (4)  005536  104014                             EMT     C$PNTB
  (4)  005540  062706  000024                     ADD     #24,SP
 3275  005544                                     PRINTB  #FMT3,#CRLCS,E.CS,#CRLBA,E.BA,#CRLDA,E.DA,#CRLMP,E.MP
  (15) 005544  013746  002242                     MOV     E.MP,-(SP)
  (14) 005550  012746  002405                     MOV     #CRLMP,-(SP)
  (13) 005554  013746  002240                     MOV     E.DA,-(SP)
  (12) 005560  012746  002373                     MOV     #CRLDA,-(SP)
  (11) 005564  013746  002236                     MOV     E.BA,-(SP)
  (10) 005570  012746  002361                     MOV     #CRLBA,-(SP)
  (9)  005574  013746  002234                     MOV     E.CS,-(SP)
  (8)  005600  012746  002333                     MOV     #CRLCS,-(SP)
  (7)  005604  012746  005734                     MOV     #FMT3,-(SP)
  (6)  005610  012746  000011                     MOV     #11,-(SP)
  (3)  005614  010600                             MOV     SP,R0
  (4)  005616  104014                             EMT     C$PNTB
  (4)  005620  062706  000024                     ADD     #24,SP
 3276  005624  000207                             RTS     PC
 3277
 3278                                    ;FORMAT STATMENTS
 3279
 3283
 3284  005626  052045  055045  022462    FMT1:    .ASCII  /%T%Z2%A:%Z2%A:%Z2/
 3285  005647     045  022524  033117    FMT17:   .ASCIZ  /%T%06%T%01/
 3286  005662  052045  052045  047045    FMT1A:   .ASCIZ  /%T%T%N/
 3287  005671     045  041101  043105    FMT2:    .ASCII  /%ABEFORE ERR%T%06/
 3288  005712  052045  047445  022466    FMT2A:   .ASCIZ  /%T%06%T%06%T%06%N/
```

```
  3289  005734  040445  052101  042440  FMT3:     .ASCIZ  /%AAT ERR    %T%06%T%06%T%06%T%06%N/
  3290  005777     045  022524  033117  FMT4:     .ASCIZ  /%T%06%T%06%N%T%06%T%06%N/
  3291  006030  052045  047445  022466  FMT5:     .ASCIZ  /%T%06%T%T%N/
  3292  006044  052045  047445  022466  FMT6:     .ASCIZ  /%T%06%T%06%A OR %06%N/
  3293  006072  047045  052045  055045  FMT7:     .ASCIZ  /%N%T%Z2%A: %Z2%A: %Z2%T%06%T/
  3294  006125     045  030517  047045  FMT7A:    .ASCIZ  /%01%N%T%A - %T%N/
  3295  006146  042045  022466  022524  FMT8:     .ASCIZ  /%D6%T%N/
  3296  006156  052045  055045  022462  FMT9:     .ASCIZ  /%T%Z2%A: %Z2%A: %Z2%T%06%T%01%N/
  3297  006214  042045  022466  020101  FMT9A:    .ASCIZ  /%D6%A WORDS BAD OUT OF %D6%A WORDS READ%N/
  3298  006266  052045  055045  022462  FMT10:  .ASCIZ    /%T%Z2%A: %Z2%A: %Z2%T%06%T%01/
  3299  006322  052045  047445  022466  FMT10A:   .ASCIZ  /%T%06%T%06%N%T%06%T%06%A AT BUS ADDRESS /
  3300  006373     045  033117  047045  FMT10B:   .ASCIZ  /%06%N/
  3301  006401     045  022524  031117  FMT11:    .ASCIZ  /%T%02%T%06%T%03/
  3302  006421     045  022524  033117  FMT12:    .ASCIZ  /%T%06%N/
  3303  006431     045  022524  033117  FMT13:    .ASCIZ  /%T%06%T%06%N/
  3304  006446  052045  055045  022464  FMT13D:   .ASCIZ  /%T%Z4%A NOW IS %Z4%N/
  3305  006473     045  022516  022524  FMT14:    .ASCIZ  /%N%T%N/
  3306  006502  047445  022466  020101  FMT14A:   .ASCIZ  /%06%A /
  3307  006511     045  000116          FMT14C:   .ASCIZ  /%N/
  3308  006514  040445  047527  042122  FMT14B:   .ASCIZ  ?%AWORD %D3%A.  S/B %06%A WAS %06%N?
  3309  006556  040445  051105  047522  FMT15:    .ASCIZ  /%AERROR(S) SET: %T%N%ARECOVERY BEING ATTEMPTED/
  3310  006634  040445  047516  020124  FRMT16:   .ASCIZ  /%ANOT TESTING CS= %06%A DR= %01%N/
  3311  006676  047045  052045     000  FMT18:    .ASCIZ  /%N%T/
  3312  006703     045  022516  022516  FMTS1:    .ASCIZ  /%N%N%S10%A*** RL01 PERFORMANCE REPORT ***%N%N/
  3313  006761     045  020101  051040  FMTS1A:   .ASCIZ  /%A  RUNNING%N/
  3314  006777     045  020101  042040  FMTS1B:   .ASCIZ  /%A  DROPPED  %Z2%A: %Z2%N/
  3315  007030  052045  047445  022465  FMTS2:    .ASCIZ  /%T%05%05%N/
  3316  007043     045  051501  042505  FMTS2A:   .ASCIZ  /%ASEEKS:  %D6%Z3%N%ABITS READ:     %D6%Z4%Z4%A (*16)%N/
  3317  007130  040445  044502  051524  FMTS2B:   .ASCIZ  /%ABITS WRITTEN:  %D6%Z4%Z4%A (*16)%N/
  3318  007174  047045  040445  051105  FMTS3:    .ASCIZ  /%N%AERRORS%N%ADRIVE:  %D6%A SEEK:  %D6%A TRACK:  %D6%A DATA:   %D6%N/
  3319  007275     045  044101  051101  FMTS3A:   .ASCIZ  /%AHARD:   %D6%A SOFT:  %D6%N/
  3320  007330  040445  041504  035113  FMTS4:    .ASCIZ  /%ADCK:    %D6%A HCRC:  %D6%A NXM:    %D6%A HNF:   %D6%N/
  3321  007414  040445  046104  035124  FMTS5:    .ASCIZ  /%ADLT:    %D6%A OPI:   %D6%N%N/
  3322
  3326
  3327
  3328
  3329
  3330          007452                  .EVEN
  3331
  3332  007452                                    ENDMOD
  3333
  3334  007452                          BGNMOD  HPTCODE
  3335
  3336  007452                          BGNHW
   (3)  007452  000005                          .WORD   L10013-L$HW/2
  3337
  3338  007454  174400                          .WORD   174400
  3339  007456  000330                          .WORD   330
  3340  007460  000240                          .WORD   240
  3341  007462  000000                          .WORD   0
  3342  007464  000001                          .WORD   1
  3343
  3344  007466                          ENDHW
   (3)  007466                          L10013:
  3345
```

```
 3346  007466                       ENDMOD
 3347
 3348                       .SBTTL  SOFTWARE PARAMETERS
 3349
 3350  007466               BGNMOD  SPTCODE
 3351
 3352  007466               BGNSW
  (3)  007466  000037               .WORD   L10014-L$SW/2
 3353
 3354  007470  000001       LIMIT:  .WORD   1                    ;RETRY LIMIT
 3355  007472  000003       ERLMT:  .WORD   3                    ;ERROR LIMIT
 3356  007474  000003       SELMT:  .WORD   3                    ;SEEK ERROR LIMIT
 3357  007476  060650       DALMT:  .WORD   25000.               ;DATA XFER LIMIT (*(10*3)) (BITS)
 3358  007500  023420       SKLMT:  .WORD   10000.               ;SEEK LIMIT
 3359  007502  000170       TYINT:  .WORD   120.                 ;TIME INTERVAL BETW/ STATISTICAL REPORT
 3360  007504  000030       CMRD:   .WORD   24.                  ;COMPARE ON READ
 3361  007506  000003       DELMT:  .WORD   3                    ;ERRORS TO REPORT ON DATA COMPARE
 3362  007510  000000       XCHFLG: .WORD   0                    ;CHANGE OTHER PARAMETERS
 3363  007512  002400       T.MXB:  .WORD   1280.                ;MAXIMUM R/W TRANSFER BUFFER
 3364  007514  000100       T.MXH:  .WORD   100                  ;MAXIMUM HEAD SELECT
 3365  007516  000000       T.MNH:  .WORD   0                    ;MINIMUM HEAD SELECT
 3366  007520  077600       T.MXC:  .WORD   77600                ;MAXIMUM CYLINDER
 3367  007522  000000       T.MNC:  .WORD   0                    ;MINIMUM CYLINDER
 3368  007524  000047       T.MXS:  .WORD   39                   ;MAXIMUM SECTOR
 3369  007526  000000       T.MNS:  .WORD   0                    ;MINIMUM SECTOR
 3370  007530  000001       T.DCK:  .WORD   1                    ;DATA DUMP ON DATA CHECK ERROR
 3371  007532  000001       T.DRP:  .WORD   1                    ;DROP ON LIMIT REACHED
 3372  007534  000003       T.MNB:  .WORD   3                    ;MINIMUM BUFFER TRANSFER SIZE
 3373  007536  000012       SFLMT:  .WORD   10.                  ;SOFT ERROR LIMIT
 3374  007540  000000       T.STA:  .WORD   0                    ;DROP DRIVE ON PERFORMANCE REACHED
 3375  007542  000003       DRLMT:  .WORD   3                    ;DRIVE ERROR L!MIT
 3376  007544  000000       T.ROF:  .WORD   0                    ;READ ONLY FLAG
 3377  007546  000001       T.RAN:  .WORD   1                    ;RANDOM SELECT OF PATTERNS
 3378  007550  000004       T.PAT:  .WORD   4                    ;ONLY ONE PATTERN  4 = WORST CASE
 3379  007552  000001       T.SLT:  .WORD   1                    ;SEEK RETRY LIMIT
 3380  007554  000200       T.CLT:  .WORD   128.                 ;NUMBER OF ERRORS ON DCK DUMP
 3381  007556  000000       T.AUT:  .WORD   0                    ;AUTO ON START UP
 3382  007560  000000       T.STIP: .WORD   0                    ;RESTRICT BUFFER SIZE
 3383  007562  000001       T.WCK:  .WORD   1                    ;DO WRITE CHECK
 3384  007564  000012       T.DCD:  .WORD   10.
 3385
 3386  007566               ENDSW
  (3)  007566               L10014:
 3387
 3388  007566                       ENDMOD
 3389
 3390  007566               BGNMOD  DSPCODE
 3391
 3392  007566               DISPATCH        1
  (4)  007566  000001               .WORD   1
  (6)  007570  012320               .WORD   T1
 3393
 3394  007572                       ENDMOD
 3395
 3396                       .SBTTL  STATISTIC CODE
 3397
```

```
  3398  007572                          BGNMOD  RPTCODE
  3399
  3400  007572                          BGNRPT
  3401
  3402
  3403  007572                                  PRINTS  #FMTS1                  ;PRINT STATISTICAL HEADER
   (7)  007572  012746  006703                  MOV     #FMTS1,-(SP)
   (6)  007576  012746  000001                  MOV     #1,-(SP)
   (3)  007602  010600                          MOV     SP,R0
   (4)  007604  104016                          EMT     C$PNTS
   (4)  007606  062706  000004                  ADD     #4,SP
  3404
  3405  007612  010446                          MOV     R4,-(SP)                ;SAVE PRESENT VALUE OF R4
  3406
  3407  007614  012704  024752                  MOV     #DRBUF,R4               ;START OF DRIVE BUFFER
  3408  007620  005764  000104          1$:     TST     DCS(R4)                 ;IS THERE A DRIVE?
  3409  007624  001402                          BEQ     2$                      ;NO, GET NEXT ONE
  3410
  3411  007626  004737  011634                  JSR     PC,REPORT               ;TYPE OUT SUMMARY
  3412
  3413  007632  062704  000122          2$:     ADD     #FRPOS+2,R4             ;NEXT DRIVE
  3414  007636  020427  026172                  CMP     R4,#ENDBUF              ;AT THE END?
  3415  007642  001366                          BNE     1$                      ;NO, TRY NEXT
  3416
  3417  007644  012604                          MOV     (SP)+,R4                ;RESTORE R4
  3418
  3419
  3420  007646                          ENDRPT
   (3)  007646                          L10015:
   (3)  007646  104025                          EMT     C$RPT
  3421
  3422  007650                                  ENDMOD
  3423
  3424                                  .SBTTL  INITIALIZATION CODE
  3425
  3426  007650                          BGNMOD  INITCODE                        ;START OF INITIALIZE CODE
  3427
  3428  007650                                  BGNINIT
  3429
  3430  007650                                  SETPRI  #340                    ;PRIORITY TO SEVEN
   (3)  007650  012700  000340                  MOV     #340,R0
   (3)  007654  104041                          EMT     C$SPRI
  3431
  3432
  3433  007656  005037  000050                  CLR     OPFLG
  3434  007662  005037  002312                  CLR     INCALL
  3435  007666  005037  002266                  CLR     STFLG
  3436  007672  005037  002270                  CLR     CNTFLG                  ;CLEAR CONT
  3437  007676                                  READEF  #EF.PWR
   (3)  007676  012700  000034                  MOV     #EF.PWR,R0
   (3)  007702  104050                          EMT     C$REFG
  3438  007704                                  BNCOMPLETE      3$
   (2)  007704  103076                          BCC     3$
  3439  007706  005237  002262                  INC     PWRFLG          ;INDICATE POWER FAIL
  3440  007712  012704  024752                  MOV     #DRBUF,R4
  3441  007716  012702  000001                  MOV     #1,R2
  3442  007722  130237  002120          11$:    BITB     R2,DRUT
  3443  007726  001446                          BEQ      13$
  3444  007730  016400  000106                  MOV      DRSEL(R4),R0
  3445  007734  052700  000200                  BIS      #200,R0
  3446  007740  010074  000104                  MOV      R0,@DCS(R4)
  3447  007744  012701  000074                  MOV      #60.,R1
  3448  007750  032774  000001  000104  12$:    BIT      #1,@DCS(R4)
  3449  007756  001014                          BNE      15$
  3450  007760                                  WAITMS   #10.
   (3)  007760  012700  000012                  MOV      #10.,R0
   (3)  007764  104026                          EMT      C$WTM
  3451  007766  005301                          DEC      R1
  3452  007770  001367                          BNE      12$
  3453
  3454  007772  012737  003672  002116          MOV      #NOPWR,WHY
  3455  010000  004537  020144                  JSR      R5,DRDRV
  3456  010004  000137  010044                  JMP      13$
  3457
  3458  010010  004537  021062          15$:    JSR      R5,ISDRST
  3459  010014  004537  022374                  JSR      R5,HDHOME
  3460  010020  005064  000056                  CLR      PRFLGS(R4)
  3461  010024  005064  000036                  CLR      RETRY(R4)
  3462  010030  005064  000076                  CLR      DOWCK(R4)
  3463  010034  005064  000052                  CLR      RTYPE(R4)
  3464  010040  005064  000114                  CLR      RSEEK(R4)
  3465  010044  062704  000122          13$:    ADD      #PRPOS+2,R4
  3466  010050  106302                          ASLB     R2
  3467  010052  103323                          BCC      11$
  3468  010054  005737  002250                  TST      SYSCLK
  3469  010060  001406                          BEQ      4$
  3470  010062                                  CLKON    #1
   (3)  010062  012700  000001                  MOV      #1,R0
   (3)  010066  104034                          EMT      C$KWON
  3471  010070                                  REQTIM   R0
   (3)  010070  104045                          EMT      C$REQTIM
  3472  010072  010037  002224                  MOV      R0,LSTTIM
  3473  010076  000137  011046          4$:     JMP      POWER
  3474  010102                          3$:     READEF   #EF.CONTINUE                    ;CONTINUE FROM CONSOLE?
   (3)  010102  012700  000036                  MOV      #EF.CONTINUE,R0
   (3)  010106  104050                          EMT      C$REFG
  3475  010110                                  BNCOMPLETE        1$                   ;NO, CONTINUE W/ INIT CODE
   (2)  010110  103004                          BCC      1$
  3476
  3477  010112  005237  002270                  INC      CNTFLG                          ;YES SET CONT FLAG, GO TO END OF INIT
  3478  010116  000137  010450                  JMP      END
  3479
  3480  010122  004537  023616          1$:     JSR      R5,CLEAR                        ;CLEAR ALL DRIVE BUFFERS
  3481  010126  012737  176543  002124          MOV      #176543,HINUM                   ;PRIME RANDOM GENERATOR
  3482  010134  012737  123456  002126          MOV      #123456,LONUM                   ;  "       "       "
  3483  010142  012700  002134          2$:     MOV      #CNTLR1,R0                      ;CLEAR FLAGS
  3484  010146  005020                  CLRDAT: CLR      (R0)+                           ;
  3485  010150  020027  002270                  CMP      R0,#STFLG+2                     ;MASS CLEAR
  3486  010154  001374                          BNE      CLRDAT                          ;
  3487
  3488  010156  012704  024752                  MOV      #DRBUF,R4                       ;SETUP UP DRIVE BUFFER POINTER
  3489  010162  012702  023724                  MOV      #BSEC0,R2                       ;SETUP BAD SECTOR POINTER
```

```
 3490  010166  013703  002014                 MOV     L$UNIT,R3               ;GET NUMBER OF UNITS
 3491  010172  010337  002260                 MOV     R3,UUT                  ;SAVE L$UNIT
 3492  010176  005001                         CLR     R1                      ;INIT P-TABLE
 3493  010200  005703                 1$:     TST     R3                      ;ANY    P-TABLES LEFT?
 3494  010202  001522                         BEQ     END                     ;NO,GO TO END
 3495  010204                                 GPHARD  R1,R0                   ;GET A P-TABLE
  (3)  010204  010100                         MOV     R1,R0
  (3)  010206  104042                         EMT     C$GPHRD
 3496  010210                                 BNCOMPLETE      12$
  (2)  010210  103110                         BCC     12$
 3497  010212  012037  002144                 MOV     (R0)+,BCSR              ;GET CSR
 3498  010216  012037  002146                 MOV     (R0)+,BVEC              ;GET VECTOR
 3499  010222  012037  002150                 MOV     (R0)+,BPRIOR            ;GET BPRIOR
 3500  010226  011037  002152                 MOV     (R0),BDRSEL             ;GET DRIVE
 3501  010232  005737  002134                 TST     CNTLR1                  ;DO WE HAVE CSR 1 YET?
 3502  010236  001011                         BNE     2$                      ;YES,THEN SEE IF IT'S IT
 3503  010240  013737  002150  002212         MOV     BPRIOR,PRIOR1
 3504  010246  013737  002144  002134         MOV     BCSR,CNTLR1             ;NO,MAKE THIS ONE CSR 1
 3505  010254  013737  002146  002206         MOV     BVEC,VECT1              ;MAKE THIS VECTOR VECT1
 3506  010262  023737  002144  002134 2$:     CMP     BCSR,CNTLR1             ;IS THIS CSR CNTLR1?
 3507  010270  001012                         BNE     5$                      ;NO,GO CHECK AGAINST #2
 3508  010272  023737  002146  002206         CMP     BVEC,VECT1              ;IS VECTOR PROPER?
 3509  010300  001050                         BNE     10$                     ;NO, REPORT ERROR
 3510  010302  012737  002252  002164         MOV     #BUF1,TEMP1             ;FIRST CONTROLLER/FIRST BUFFER
 3511  010310  004537  011424                 JSR     R5,FILINF               ;FILL BUFFER
 3512  010314  000450                         BR      11$                     ;GO GET NEXT P-TABLE
 3513  010316  005737  002136         5$:     TST     CNTLR2                  ;HAVE WE GOT CSR #2 YET?
 3514  010322  001015                         BNE     6$                      ;YES, CHECK THIS ONE AGAINST IT
 3515  010324  023737  002206  002144         CMP     VECT1,BCSR              ;IS THIS VECTOR SAME AS CNTLR1
 3516  010332  001433                         BEQ     10$                     ;IFSO, DON'T ALLOW IT
 3517  010334  013737  002144  002136         MOV     BCSR,CNTLR2             ;MAKE THIS ONE CSR 2
 3518  010342  013737  002146  002210         MOV     BVEC,VECT2              ;SETUP SECOND VECTOR
 3519  010350  013737  002150  002214         MOV     BPRIOR,PRIOR2
 3520  010356  023737  002144  002136 6$:     CMP     BCSR,CNTLR2             ;IS THIS CSR # 2?
 3521  010364  001016                         BNE     10$                     ;NO, WELL WE DON'T ALLOW 3
 3522  010366  023737  002146  002210         CMP     BVEC,VECT2              ;DOES IT HAVE PROPER VECTOR
 3523  010374  001012                         BNE     10$                     ;NO, GO REPORT ERROR
 3524  010376  023737  002210  002206         CMP     VECT2,VECT1             ;IS VECTOR OF FIRST EQUAL TO
 3525  010404  001406                         BEQ     10$                     ;VECTOR OF SECOND, YES REPORT ERROR
 3526  010406  012737  002254  002164         MOV     #BUF2,TEMP1             ;OTHER CNTLR/OTHER BUFFER
 3527  010414  004537  011424                 JSR     R5,FILINF               ;LOAD BUFFER
 3528  010420  000406                         BR      11$                     ;NEXT
 3529  010422                         10$:    ERRDF   160.,ILLEG,ERR10        ;BAD P-TABLE
  (3)  010422  104462                         TRAP    T$ERCODE
  (5)  010424  000240                         .WORD   160
  (5)  010426  003560                         .WORD   ILLEG
  (5)  010430  005100                         .WORD   ERR10
 3530  010432  005064  000104         12$:    CLR     DCS(R4)
 3531  010436  005201                 11$:    INC     R1                      ;POINT TO NEXT
 3532  010440  005303                         DEC     R3                      ;DOWN COUNT
 3533  010442  062702  000040                 ADD     #32.,R2                 ;NEXT BAD SECTOR FILE
 3534  010446  000654                         BR      1$                      ;DO WHILE
 3535
 3536
 3537  010450                         END:
 3538
```

```
      010450  012737  177770  002122                  MOV     #177770,SYSMSK                  ;SETUP FOR EIGHT DRIVES
      010456  023727  002260  000004                  CMP     UUT,#4                          ;MORE THAN FOUR
      010464  003012                                  BGT     2$                              ;YES, THEN MASK IS OKAY
      010466  052737  000004  002122                  BIS     #4,SYSMSK                       ;SETUP FOR FOUR DRIVES
      010474  023727  002260  000002                  CMP     UUT,#2                          ;MORE THAN TWO
      010502  003003                                  BGT     2$                              ;YES, IT'S OKAY
      010504  052737  000002  002122                  BIS     #2,SYSMSK                       ;SET FOR ONE OR TWO
      010512                                  2$:     READEF  #EF.START                       ;START COMMAND
 (3)  010512  012700  000040                          MOV     #EF.START,R0
 (3)  010516  104050                                  EMT     C$REFG
      010520                                          BNCOMPLETE      RESTART                 ;NO, CHK RESTART
 (2)  010520  103002                                  BCC     RESTART
      010522  005237  002266                          INC     STFLG                           ;SET START INDICATOR

      010526  005737  002270                  RESTART:        TST     CNTFLG                  ;CONTINUING
      010532  001024                                  BNE     3$                              ;YES GO TO 3$



                                              ;LET'S CREATE INTERNAL BITMAP

      010534  012701  000001                          MOV     #1,R1                           ;BIT MASK
      010540  105037  002121                          CLRB    DRPRS                           ;CLEAR OUT DRIVES PRESENT
      010544  012704  024752                          MOV     #DRBUF,R4                       ;START OF DRIVE BUFFERS
      010550  005764  000104                  1$:     TST     DCS(R4)                         ;ANY CSR?
      010554  001402                                  BEQ     2$                              ;NO, NO DRIVE THEN
      010556  150137  002121                          BISB    R1,DRPRS                        ;INDICATE DRIVE IN BITMAP
      010562  006301                          2$:     ASL     R1                              ;NEXT POSITION
      010564  062704  000122                          ADD     #PRPOS+2,R4                     ;NEXT DRIVE BUFFER
      010570  022704  026172                          CMP     #ENDBUF,R4                      ;DONE
      010574  001365                                  BNE     1$                              ;NO

      010576  113737  002121  002120                  MOVB    DRPRS,DRUT                      ;SET UP DRIVES UNDER TEST

      010604                                  3$:

      010604                                          SETVEC  VECT1,#INTR1,PRIOR1             ;SET CONTROLLER 1'S VECTOR
 (7)  010604  013746  002212                          MOV     PRIOR1,-(SP)
 (6)  010610  012746  014152                          MOV     #INTR1,-(SP)
 (5)  010614  013746  002206                          MOV     VECT1,-(SP)
 (4)  010620  012746  000003                          MOV     #3,-(SP)
 (3)  010624  104037                                  EMT     C$SVEC
 (2)  010626  062706  000010                          ADD     #10,SP

      010632  005737  002136                          TST     CNTLR2                          ;RUNNING TWO CONTROLLERS?
      010636  001413                                  BEQ     4$                              ;NO

      010640                                          SETVEC  VECT2,#INTR2,PRIOR2             ;YES SET CONTROLLER 2'S VECTOR
 (7)  010640  013746  002214                          MOV     PRIOR2,-(SP)
 (6)  010644  012746  014162                          MOV     #INTR2,-(SP)
 (5)  010650  013746  002210                          MOV     VECT2,-(SP)
 (4)  010654  012746  000003                          MOV     #3,-(SP)
 (3)  010660  104037                                  EMT     C$SVEC
 (2)  010662  062706  000010                          ADD     #10,SP

      010666  005737  002270                  4$:     TST     CNTFLG                          ;CONTINUE?
      010672  001412                                  BEQ     FINDBF                          ;NO, GO PAST RESTART OF CLOCK
 3580
 3581  010674  005737  002250                  TST     SYSCLK                        ;DO WE HAVE SYSTEM CLOCK
 3582  010700  001462                          BEQ     POWER                         ;NO
 3583
 3584  010702                                  CLKON   #1                            ;TURN CLK ON
  (3)  010702  012700  000001                  MOV     #1,R0
  (3)  010706  104034                          EMT     C$KWON
 3585  010710                                  REQTIM  R0                            ;REQUEST TIME
  (3)  010710  104045                          EMT     C$REQTIM
 3586  010712  010037  002224                  MOV     R0,LSTTIM                     ;MAKE IT PRESENT TIME
 3587  010716  000453                          BR      POWER                         ;GO TO END
 3588
 3589
 3590  010720  012703  000050        FINDBF:   MOV     #40.,R3           ;MAXIMUM SECTOR IS 40
 3591  010724  005001                          CLR     R1                            ;START WC AT ZERO
 3592  010726  005737  002136                  TST     CNTLR2                        ;TWO CONTROLLERS????
 3593  010732  001402                          BEQ     1$                            ;NO, START WC AT 5120
 3594  010734  012701  000024                  MOV     #20.,R1                       ;20 256 WORD BUFFERS
 3595  010740  062701  000024        1$:       ADD     #20.,R1                       ;WC TO 5120 PLUS 5120
 3596  010744                        2$:       BUFFER  R1,R2             ;GET BUFFER IF AVAILABLE
  (3)  010744  010100                          MOV     R1,R0
  (3)  010746  104030                          EMT     C$BUFF
  (3)  010750  010002                          MOV     R0,R2
 3597  010752                                  BCOMPLETE       4$                    ;WAS AVAILABLE, THEN BR
  (2)  010752  103411                          BCS     4$
 3598  010754  005737  002136                  TST     CNTLR2                        ;TWO CONTROLLERS???
 3599  010760  001401                          BEQ     3$                            ;NO
 3600  010762  005301                          DEC     R1                            ;ONE 256 WORD BUFFER LESS
 3601  010764  005301                3$:       DEC     R1                            ;ONE MORE LESS
 3602  010766  162703  000002                  SUB     #2,R3
 3603  010772  001364                          BNE     2$                            ;IF NOT ZERO GO BACK
 3604
 3605  010774  000000                          HALT
 3606
 3607  010776  042701  177400        4$:       BIC     #177400,R1
 3608  011002  000301                          SWAB    R1
 3609  011004  010237  002252                  MOV     R2,BUF1                       ;GET BUFFER FOR FIRST CONTROLLER
 3610  011010  005737  002136                  TST     CNTLR2                        ;TWO CONTROLLERS??
 3611  011014  001404                          BEQ     5$                            ;NO
 3612  011016  060102                          ADD     R1,R2                         ;SECOND'S BUFFER
 3613  011020  010237  002254                  MOV     R2,BUF2
 3614  011024  006201                          ASR     R1                            ;CORRECT WORD COUNT
 3615  011026  010137  002256        5$:       MOV     R1,MAXWC                      ;MAX WORD COUNT
 3616
 3617
 3618
 3619  011032                        7$:       CLKON   #1                            ;TURN CLOCK ON?
  (3)  011032  012700  000001                  MOV     #1,R0
  (3)  011036  104034                          EMT     C$KWON
 3620  011040                                  BNCOMPLETE      POWER                 ;WAS THERE A CLOCK?
  (2)  011040  103002                          BCC     POWER
 3621
 3622  011042  005237  002250                  INC     SYSCLK                        ;YES, SET FLAG FOR ONE!
 3623
 3624  011046                        POWER:
 3625
```

```


      011046                                           ENDINIT
 (3)  011046                                   L10016.
 (3)  011046  104011                                   EMT      C$INIT

      011050                                           ENDMOD

      011050                                   BGNMOD  CLNCODE


      011050                                           BGNCLN

      011050                                           SETVEC   ERRVEC,#TRPHAN,#340
 (7)  011050  012746  000340                           MOV      #340,-(SP)
 (6)  011054  012746  011626                           MOV      #TRPHAN,-(SP)
 (5)  011060  013746  002302                           MOV      ERRVEC,-(SP)
 (4)  011064  012746  000003                           MOV      #3,-(SP)
 (3)  011070  104037                                   EMT      C$SVEC
 (2)  011072  062706  000010                           ADD      #10,SP
      011076                                           SETPRI   #FRI00                    ;PRIORITY TO ZERO
 (3)  011076  012700  000000                           MOV      #PRI00,R0
 (3)  011102  104041                                   EMT      C$SPRI

      011104  032777  000200  171022  1$:              BIT      #CRDY,@CNTLR1             ;WAIT FOR CONTROLLER TO FINISH
      011112  001774                                   BEQ      1$                        ;
      011114  042777  000100  171012                   BIC      #INTEN,@CNTLR1            ;CLEAR INTERRUPT IF PENDING
      011122                                           CLRVEC   VECT1                     ;RELEASE VECTOR OF FIRST CONTROLLER
 (3)  011122  013700  002206                           MOV      VECT1,R0
 (3)  011126  104036                                   EMT      C$CVEC

      011130  005737  002136                           TST      CNTLR2                    ;TWO CONTROLLERS
      011134  001412                                   BEQ      3$                        ;NO

      011136  032777  000200  170772  2$:              BIT      #CRDY,@CNTLR2             ;WAIT FOR OTHER CONTROLLER TO FINISH
      011144  001774                                   BEQ      2$                        ;
      011146  042777  000100  170762                   BIC      #INTEN,@CNTLR2            ;CLEAR OUT INTERRUPT ENABLE
      011154                                           CLRVEC   VECT2                     ;YES, WELL RELEASE IT'S VECTOR
 (3)  011154  013700  002210                           MOV      VECT2,R0
 (3)  011160  104036                                   EMT      C$CVEC

      011162  005037  002312          3$:              CLR      INCALL
      011166  005037  002310                           CLR      OPCALL
      011172                                           CLRVEC   ERRVEC
 (3)  011172  013700  002302                           MOV      ERRVEC,R0
 (3)  011176  104036                                   EMT      C$CVEC

      011200                                           ENDCLN
 (3)  011200                                   L10017:
 (3)  011200  104012                                   EMT      C$CLEAN

      011202                                           ENDMOD


      011202                                   BGNMOD  ADDCODE

```

```
  3664  011202                          BGNAU
  3665
  3666  011202  012704  024752                   MOV     #DRBUF,R4               ;START OF DRIVE BUFFERS
  3667  011206  012701  000001                   MOV     #1,R1                   ;MASK TO FIND DRIVE
  3668  011212  010002                           MOV     R0,R2                   ;SAVE WHICH TO FIND
  3669  011214  005700                  1$:      TST     R0                      ;THIS ONE
  3670  011216  001405                           BEQ     2$                      ;YES
  3671  011220  062704  000122                   ADD     #PRPOS+2,R4             ;NEXT
  3672  011224  006301                           ASL     R1                      ;NEXT MASK
  3673  011226  005300                           DEC     R0
  3674  011230  000771                           BR      1$
  3675  011232  150137  002120          2$:      BISB    R1,DRUT                 ;INSERT IN DRIVE UNDER TEST
  3676  011236                                   GPHARD  R2,R1
   (3)  011236  010200                           MOV     R2,R0
   (3)  011240  104042                           EMT     C$GPHRD
   (3)  011242  010001                           MOV     R0,R1
  3677  011244  011164  000104                   MOV     (R1),DCS(R4)
  3678  011250  012700  000100                   MOV     #SERNM1,R0              ;SETUP TO CLEAR STATS
  3679  011254  006200                           ASR     R0
  3680  011256  005024                  4$:      CLR     (R4)+
  3681  011260  005300                           DEC     R0
  3682  011262  001375                           BNE     4$
  3683  011264                          5$:
  3684
  3685  011264                          ENDAU
   (3)  011264                          L10020:
   (3)  011264  104054                           EMT     C$AU
  3686
  3687  011266                          ENDMOD
  3688
  3689  011266                          BGNMOD  DROPCODE
  3690
  3691  011266                          BGNDU
  3692
  3693  011266  005737  002312                   TST     INCALL
  3694  011272  001015                           BNE     3$
  3695  011274  012704  024752                   MOV     #DRBUF,R4
  3696  011300  005700                  2$:      TST     R0
  3697  011302  001404                           BEQ     1$
  3698  011304  005300                           DEC     R0
  3699  011306  062704  000122                   ADD     #PRPOS+2,R4
  3700  011312  000772                           BR      2$
  3701
  3702  011314  012737  003277  002116  1$:      MOV     #REQ,WHY
  3703  011322  004537  020140                   JSR     R5,ODRDRV
  3704  011326                          3$:
  3705
  3706
  3707  011326                          ENDDU
   (3)  011326                          L10021:
   (3)  011326  104055                           EMT     C$DU
  3708
  3709  011330                          ENDMOD
  3710
  3711                                   SBTTL  GLOBAL SUBROUTINES
  3712
```

```
 3713  011330                          BGNMOD  GLBSUB
 3714
 3715  011330  012701  000010          SETWCK: MOV     #8.,R1
 3716  011334  012702  024752                  MOV     #DRBUF,R2
 3717  011340  026462  000104  000104  1$:     CMP     DCS(R4),DCS(R2)
 3718  011346  001002                          BNE     2$
 3719  011350  010462  000076                  MOV     R4,DOWCK(R2)
 3720  011354  062702  000122          2$:     ADD     #PRPOS+2,R2
 3721  011360  005301                          DEC     R1
 3722  011362  001366                          BNE     1$
 3723  011364  000205                          RTS     R5
 3724
 3725  011366  012701  000010          CLRWCK: MOV     #8.,R1
 3726  011372  012702  024752                  MOV     #DRBUF,R2
 3727  011376  026462  000104  000104  1$:     CMP     DCS(R4),DCS(R2)
 3728  011404  001002                          BNE     2$
 3729  011406  005062  000076                  CLR     DOWCK(R2)
 3730  011412  062702  000122          2$:     ADD     #PRPOS+2,R2
 3731  011416  005301                          DEC     R1
 3732  011420  001366                          BNE     1$
 3733  011422  000205                          RTS     R5
 3734
 3735                                  ;
 3736                                  ;ROUTINE TO FILL BUFFERS WITH INFO
 3737
 3738  011424  013764  002152  000106  FILINF: MOV     BDRSEL,DRSEL(R4)        ;SET DRIVE SELECT BITS
 3739  011432  013764  002144  000104          MOV     BCSR,DCS(R4)            ;SET CSR
 3740  011440  013764  002164  000110          MOV     TEMP1,BBA(R4)           ;SET R/W BUFFER
 3741  011446  010264  000112                  MOV     R2,BSECPT(R4)           ;SETUP BAD SECTOR POINTER
 3742  011452  005737  007556                  TST     T.AUT                   ;DO WE AUTOSIZE?
 3743  011456  001460                          BEQ     1$                      ;NO, SKIP
 3744
 3745  011460  005037  002264                  CLR     TRPFLG                  ;CLEAR  TRAP FLAG
 3746  011464                                  SETVEC  ERRVEC,#TRPHAN,#340     ;SETUP TO CATCH TRAP
  (7)  011464  012746  000340                  MOV     #340,-(SP)
  (6)  011470  012746  011626                  MOV     #TRPHAN,-(SP)
  (5)  011474  013746  002302                  MOV     ERRVEC,-(SP)
  (4)  011500  012746  000003                  MOV     #3,-(SP)
  (3)  011504  104037                          EMT     C$SVEC
  (2)  011506  062706  000010                  ADD     #10,SP
 3747  011512  005774  000104                  TST     @DCS(R4)
 3748  011516  005737  002264                  TST     TRPFLG                  ;DID TRAP OCCUR
 3749  011522  001012                          BNE     3$                      ;YES IGNORE DRIVE
 3750  011524  016400  000106                  MOV     DRSEL(R4),R0            ;YES, FIND OUT IF DRIVE
 3751  011530  052700  000200                  BIS     #200,R0                 ;HAS DRIVE READY POSTED
 3752  011534  010074  000104                  MOV     R0,@DCS(R4)
 3753  011540  032774  000001  000104          BIT     #1,@DCS(R4)             ;IS DRIVE READY HIGH?
 3754  011546  001021                          BNE     2$                      ;YES, CHECK NEXT
 3755
 3756  011550                          3$:     PRINTF  #FRMT16,DCS(R4),<B,DRSEL+1(R4)>
  (9)  011550  005046                          CLR     -(SP)
  (9)  011552  156416  000107                  BISB    DRSEL+1(R4),(SP)
  (8)  011556  016446  000104                  MOV     DCS(R4),-(SP)
  (7)  011562  012746  006634                  MOV     #FRMT16,-(SP)
  (6)  011566  012746  000003                  MOV     #3,-(SP)
  (3)  011572  010600                          MOV     SP,R0
```

```
   (4)  011574  104017                              EMT     C$PNTF
   (4)  011576  062706  000010                      ADD     #10,SP
  3757
  3758  011602  005337  002260                      DEC     UUT                             ;ONE LESS DRIVE NOW
  3759  011606  005064  000104                      CLR     DCS(R4)                         ;TAKE DRIVE OUT OF BUFFER
  3760  011612                             2$:      CLRVEC  ERRVEC                          ;RELEASE THE VECTOR
   (3)  011612  013700  002302                      MOV     ERRVEC,R0
   (3)  011616  104036                              EMT     C$CVEC
  3761  011620  062704  000122             1$:      ADD     #PRPOS+2,R4                     ;UPDATE POINTER
  3762  011624  000205                              RTS     R5
  3763  011626  005237  011626             TRPHAN:  INC     TRPHAN
  3764  011632  000002                              RTI
  3765
  3766                                     ;ROUTINE TO PRINT STATISTICAL REPORT OF DRIVE(S)
  3767
  3768  011634                             REPORT:
  3769
  3770  011634                                      PRINTS  #FMT1,#TIME,HOUR,MINUTE,SECOND,#MRLCS,DCS(R4),#DRNM,<B,DRSEL+1(R4)>
  (15)  011634  005046                              CLR     -(SP)
  (15)  011636  156416  000107                      BISB    DRSEL+1(R4),(SP)
  (14)  011642  012746  003620                      MOV     #DRNM,-(SP)
  (13)  011646  016446  000104                      MOV     DCS(R4),-(SP)
  (12)  011652  012746  002323                      MOV     #MRLCS,-(SP)
  (11)  011656  013746  002226                      MOV     SECOND,-(SP)
  (10)  011662  013746  002230                      MOV     MINUTE,-(SP)
   (9)  011666  013746  002232                      MOV     HOUR,-(SP)
   (8)  011672  012746  002314                      MOV     #TIME,-(SP)
   (7)  011676  012746  005626                      MOV     #FMT1,-(SP)
   (6)  011702  012746  000011                      MOV     #11,-(SP)
   (3)  011706  010600                              MOV     SP,R0
   (4)  011710  104016                              EMT     C$PNTS
   (4)  011712  062706  000024                      ADD     #24,SP
  3771
  3772  011716  005764  000070                      TST     DPHOUR(R4)                      ;DO WE HAVE ANY DROPPED TIME
  3773  011722  001417                              BEQ     1$                              ;NO, THEN PRINT RUNNING
  3774
  3775  011724                                      PRINTS  #FMTS1B,<B,DPHOUR(R4)>,<B,DPMIN(R4)>
   (9)  011724  005046                              CLR     -(SP)
   (9)  011726  156416  000071                      BISB    DPMIN(R4),(SP)
   (8)  011732  005046                              CLR     -(SP)
   (8)  011734  156416  000070                      BISB    DPHOUR(R4),(SP)
   (7)  011740  012746  006777                      MOV     #FMTS1B,-(SP)
   (6)  011744  012746  000003                      MOV     #3,-(SP)
   (3)  011750  010600                              MOV     SP,R0
   (4)  011752  104016                              EMT     C$PNTS
   (4)  011754  062706  000010                      ADD     #10,SP
  3776  011760  000410                              BR      2$
  3777
  3778  011762                             1$:      PRINTS  #FMTS1A
   (7)  011762  012746  006761                      MOV     #FMTS1A,-(SP)
   (6)  011766  012746  000001                      MOV     #1,-(SP)
   (3)  011772  010600                              MOV     SP,R0
   (4)  011774  104016                              EMT     C$PNTS
   (4)  011776  062706  000004                      ADD     #4,SP
  3779  012002                             2$:
  3780  012002                                      PRINTS  #FMTS2,#CART,SERNM2(R4),SERNM1(R4)
```

```
   (10) 012002  016446  000100                          MOV     SERNM1(R4),-(SP)
   (9)  012006  016446  000102                          MOV     SERNM2(R4),-(SP)
   (8)  012012  012746  002430                          MOV     #CART,-(SP)
   (7)  012016  012746  007030                          MOV     #FMTS2,-(SP)
   (6)  012022  012746  000004                          MOV     #4,-(SP)
   (3)  012026  010600                                  MOV     SP,R0
   (4)  012030  104016                                  EMT     C$PNTS
   (4)  012032  062706  000012                          ADD     #12,SP
 3781   012036                                          PRINTS  #FMTS2A,SKCNT(R4),SKCNT1(R4),RXFR3(R4),RXFR2(R4),RXFR1(R4)
   (12) 012036  016446  000002                          MOV     RXFR1(R4),-(SP)
   (11) 012042  016446  000004                          MOV     RXFR2(R4),-(SP)
   (10) 012046  016446  000060                          MOV     RXFR3(R4),-(SP)
   (9)  012052  016446  000054                          MOV     SKCNT1(R4),-(SP)
   (8)  012056  016446  000000                          MOV     SKCNT(R4),-(SP)
   (7)  012062  012746  007043                          MOV     #FMTS2A,-(SP)
   (6)  012066  012746  000006                          MOV     #6,-(SP)
   (3)  012072  010600                                  MOV     SP,R0
   (4)  012074  104016                                  EMT     C$PNTS
   (4)  012076  062706  000016                          ADD     #16,SP
 3782   012102                                          PRINTS  #FMTS2B,WXFR3(R4),WXFR2(R4),WXFR1(R4)
   (10) 012102  016446  000006                          MOV     WXFR1(R4),-(SP)
   (9)  012106  016446  000010                          MOV     WXFR2(R4),-(SP)
   (8)  012112  016446  000062                          MOV     WXFR3(R4),-(SP)
   (7)  012116  012746  007130                          MOV     #FMTS2B,-(SP)
   (6)  012122  012746  000004                          MOV     #4,-(SP)
   (3)  012126  010600                                  MOV     SP,R0
   (4)  012130  104016                                  EMT     C$PNTS
   (4)  012132  062706  000012                          ADD     #12,SP
 3783   012136                                          PRINTS  #FMTS3,DERCNT(R4),SKECNT(R4),TRERR(R4),DATCER(R4)
   (11) 012136  016446  000074                          MOV     DATCER(R4),-(SP)
   (10) 012142  016446  000072                          MOV     TRERR(R4),-(SP)
   (9)  012146  016446  000016                          MOV     SKECNT(R4),-(SP)
   (8)  012152  016446  000020                          MOV     DERCNT(R4),-(SP)
   (7)  012156  012746  007174                          MOV     #FMTS3,-(SP)
   (6)  012162  012746  000005                          MOV     #5,-(SP)
   (3)  012166  010600                                  MOV     SP,R0
   (4)  012170  104016                                  EMT     C$PNTS
   (4)  012172  062706  000014                          ADD     #14,SP
 3784   012176                                          PRINTS  #FMTS3A,ERRCNT(R4),SFTCNT(R4)
   (9)  012176  016446  000014                          MOV     SFTCNT(R4),-(SP)
   (8)  012202  016446  000012                          MOV     ERRCNT(R4),-(SP)
   (7)  012206  012746  007275                          MOV     #FMTS3A,-(SP)
   (6)  012212  012746  000003                          MOV     #3,-(SP)
   (3)  012216  010600                                  MOV     SP,R0
   (4)  012220  104016                                  EMT     C$PNTS
   (4)  012222  062706  000010                          ADD     #10,SP
 3785   012226                                          PRINTS  #FMTS4,DCRCER(R4),HCRCER(R4),NXMCNT(R4),HNFERR(R4)
   (11) 012226  016446  000032                          MOV     HNFERR(R4),-(SP)
   (10) 012232  016446  000034                          MOV     NXMCNT(R4),-(SP)
   (9)  012236  016446  000024                          MOV     HCRCER(R4),-(SP)
   (8)  012242  016446  000022                          MOV     DCRCER(R4),-(SP)
   (7)  012246  012746  007330                          MOV     #FMTS4,-(SP)
   (6)  012252  012746  000005                          MOV     #5,-(SP)
   (3)  012256  010600                                  MOV     SP,R0
   (4)  012260  104016                                  EMT     C$PNTS
   (4)  012262  062706  000014                          ADD     #14,SP
```

```
3786  012266                                    PRINTS  #FMTS5,DLTCNT(R4),OPICNT(R4)
 (9)  012266  016446  000030                    MOV     OPICNT(R4),-(SP)
 (8)  012272  016446  000026                    MOV     DLTCNT(R4),-(SP)
 (7)  012276  012746  007414                    MOV     #FMTS5,-(SP)
 (6)  012302  012746  000003                    MOV     #3,-(SP)
 (3)  012306  010600                            MOV     SP,R0
 (4)  012310  104016                            EMT     C$PNTS
 (4)  012312  062706  000010                    ADD     #10,SP
3787  012316  000207                            RTS     PC
3788
3789
3790  012320                                    ENDMOD
3791
3792                                    .SBTTL  PROGRAM MAIN LOOP
3793
3794  012320                            BGNTST
3795                                    ;MAIN PROGRAM LOOP
3796                                    ;PROGRAM WILL RANDOMLY PICK ONE OF THE DRIVES TO
3797                                    ;PERFORM AN OPERATION.  WE WILL ALWAYS PICK ONE OF FOUR
3798                                    ;OR EiGHT DRIVES (ONE OR TWO CONTROLLERS) "DRUT" WILL BE
3799                                    ;CHECKED TO SEE IF DRIVE IS ON SYSTEM.  ONCE DRIVE IS PICKED
3800                                    ;THEN A FUNCTION WILL BE SELECTED RANDOMLY FOR THAT
3801                                    ;DRIVE.  FUNCTIONS OF CONTROLLER RESET, GET STATUS, SEEK, READ, WRITE
3802                                    ;WILL BE SELECTED, EACH FUNCTION WILL HAVE IT'S OWN ROUTINE
3803                                    ;TO GET PARAMETERS FOR THE DRIVE.
3804
3805  012320  005737  002262            MTEST:  TST     PWRFLG
```

```
 3807  012324  001054                    BNE     12$                         ;IF POWER FAIL SKIF
```

```
012326  012704  024752                       MOV     #DRBUF,R4                   ;GET DRIVE BUFFERS
012332  012701  000001                       MOV     #1,R1                       ;MASK
012336  130137  002120              16$:     BITB    R1,DRUT                     ;DRIVE UNDER TEST
012342  001441                               BEQ     15$                         ;NO

012344  012774  000200  000104               MOV     #200,@DCS(R4)               ;CHECK IF DRIVE THERE
012352  056474  000106  000104               BIS     DRSEL(R4),@DCS(R4)
012360  012700  000012                       MOV     #10.,R0                     ;STALL
012364  005300                      13$:     DEC     R0
012366  001376                               BNE     13$
```

```
     012370  032774  000001  000104             BIT     #DRDY,@DCS(R4)
     012376  001006                             BNE     14$

     012400  012737  002460  002116             MOV     #DNRDY,WHY
     012406  004537  020144                     JSR     R5,DRDRV
     012412  000415                             BR      15$

     012414  004537  017432           14$:      JSR     R5,RDBDSC              ;GO GET BAD SECTORS
     012420  005064  000056                     CLR     PRFLGS(R4)
     012424  005064  000076                     CLR     DOWCK(R4)
     012430  005064  000114                     CLR     RSEEK(R4)
     012434  005737  002266                     TST     STFLG
     012440  001402                             BEQ     15$

     012442  004537  021236                     JSR     R5,WRPACK


     012446  062704  000122           15$:      ADD     #PRPOS+2,R4            ;NEXT DRIVE
     012452  106301                             ASLB    R1                     ;DONE?
     012454  103330                             BCC     16$                    ;NO GO FOR NEXT ONE
     012456                           12$:      PRINTF  #FMT14,#MSTART
 (8) 012456  012746  004151                     MOV     #MSTART,-(SP)
 (7) 012462  012746  006473                     MOV     #FMT14,-(SP)
 (6) 012466  012746  000002                     MOV     #2,-(SP)
 (3) 012472  010600                             MOV     SP,R0
 (4) 012474  104017                             EMT     C$PNTF
 (4) 012476  062706  000006                     ADD     #6,SP
     012502                                     SETPRI  #0             ;PRIORITY TO ZERO
 (3) 012502  012700  000000                     MOV     #0,R0
 (3) 012506  104041                             EMT     C$SPRI
     012510  004537  021140           MAIN:     JSR     R5,RAND        ;GET A DRIVE?(LUN)
     012514  013702  002126                     MOV     LONUM,R2       ;GET THE SELECTED DRIVE (LUN)
     012520  043702  002122           PEROTH:   BIC     SYSMSK,R2      ;MASK TO DRIVES ON SYSTEM
     012524  012701  000001                     MOV     #1,R1          ;LET'S SEE IF DRIVE IS THERE
     012530  005702                   1$:       TST     R2             ;HAVE WE GOT PROPER MASK YET
     012532  001403                             BEQ     2$             ;YES, GO TO 2$
     012534  006301                             ASL     R1             ;NO, SHIFT FOR NEXT DRIVE
     012536  005302                             DEC     R2             ;DECREMENT DRIVE NUMBER
     012540  000773                             BR      1$             ;GO CHECK NEW DRIVE NUMBER
     012542  105737  002120           2$:       TSTB    DRUT           ;ANY DRIVES ON LINE
     012546  001005                             BNE     5$             ;YES, CHECK

     012550                                     ERRSF   170.,NODRIV    ;NO DRIVES
 (3) 012550  104421                             TRAP    T$ERCODE
 (5) 012552  000252                             .WORD   170
 (5) 012554  003606                             .WORD   NODRIV

     012556  000137  024744                     JMP     ENDOFPROGRAM

     012562  130137  002120           5$:       BITB    R1,DRUT        ;IS THIS DRIVE PRESENT?
     012566  001750                             BEQ     MAIN           ;NO, GO BACK TRY AGAIN

                                      ;WE NOW HAVE A DRIVE, CHECK TO SEE IF IT'S CONTROLLER
                                      ;IS FREE BEFORE WE GO ANY FURTHER


```

```

012570  004537  022252                   JSR     R5,GETSYS         ;GET PRESENT TIME OF SYSTEM
012574  023737  002222  007502           CMP     INTERVAL,TYINT    ;TIME TO PRINT REPORT
012602  002403                           BLT     6$                ;NO, PERFORM FUNCTION
012604  005037  002222                   CLR     INTERVAL          ;YES, START INTERVAL OVER

012610                                   DORPT                     ;PRINT STATISTICAL REPORT
 (3)  012610  104024                     EMT     C$DRPT

012612  012704  024752          6$:      MOV     #DRBUF,R4         ;GET START OF DRIVE BUFFERS
012616  013702  002126                   MOV     LONUM,R2          ;GET RANDOM DRIVE BACK (LUN)
012622  043702  002122                   BIC     SYSMSK,R2         ;MASK TO SYSTEM SYS
012626  005702                  3$:      TST     R2                ;DO WE HAVE BUFFER FOR THAT DRIVE
012630  001404                           BEQ     4$                ;YES, GO CHECK IT'S CONTROLLER
012632  062704  000122                   ADD     #PRPOS+2,R4       ;NO, UPDATE FOR NEXT BUFFER
012636  005302                           DEC     R2                ;DOWN COUNT DRIVE NUMBER (LUN)
012640  000772                           BR      3$                ;GO BACK AND CHECK FOR FOUND
012642  032774  000200  000104  4$:      BIT     #BIT7,@DCS(R4)    ;CONTROLLER ASSOCIATED WITH DRIVE
012650  001404                           BEQ     OTHCTL            ;BUSY, CHECK OTHER CONTROLLER
012652  032774  000100  000104           BIT     #BIT6,@DCS(R4)    ;INTERRUPT BEEN SERVICED?
012660  001421                           BEQ     TAGX              ;YES,GO DO OPERATION
012662  005737  002136          OTHCTL:  TST     CNTLR2            ;TWO ONTROLLERS?
012666  001710                           BEQ     MAIN              ;NO,FORGET IT
012670  013702  002126                   MOV     LONUM,R2          ;GET RANDOM NUMBER
012674  026437  000104  002134           CMP     DCS(R4),CNTLR1    ;WHICH CNTLR WAS BUSY
012702  001404                           BEQ     1$                ;FIRST, SKIP
012704  000261                           SEC
012706  006002                           ROR     R2
012710  000137  012520                   JMP     PEROTH
012714  000261                  1$:      SEC
012716  006102                           ROL     R2
012720  000137  012520                   JMP     PEROTH

                                ;WE CAN NOW PROCEED IN GETTING A FUNCTION AND RELATED DATA
                                ;FOR THE DRIVE RANDOMLY.  R4 HAS DRIVE BUFFER POINTER

012724  005764  000076          TAGX:    TST     DOWCK(R4)         ;WRITE CHECK NEEDED
012730  001407                           BEQ     80$               ;NO
012732  016404  000076                   MOV     DOWCK(R4),R4      ;GET ONE THAT NEEDS TO BE WRCHK'D
012736  012764  000002  000044           MOV     #WRCHK,FUNC(R4)   ;WRITE CHECK
012744  000137  014002                   JMP     ISSUE             ;ISSUE IT
012750  005764  000036          80$:     TST     RETRY(R4)         ;DOES DRIVE HAVE RETRY IN
012754  001402                           BEQ     78$               ;PROGRESS, NO CONTINUE
012756  000137  014002                   JMP     ISSUE             ;GO RETRY COMMAND

012762  005764  000114          78$:     TST     RSEEK(R4)   ;RECOVERY FROM SEEK ERROR
012766  001003                           BNE     77$               ;NO
012770  000412                           BR      GETFNC            ;NO, CONTINUE
012772  000137  013732                   JMP     RDDFNC            ;GO READ
012776  032764  000001  000056  77$:     BIT     #SKDON,PRFLGS(R4)   ;SEEK BEEN VERIFIED
013004  001002                           BNE     79$               ;NO
013006  000137  013324                   JMP     SKFNC             ;GO, TRY TO RECOVER
013012  000137  013650          79$:     JMP     RDHFNC            ;GO VERIFY SEEK

                                ;CHECK LIMITS OF ERRORS/OPERATIONS

```

```
013016  032764  000001  000056  GETFNC: BIT     #SKDON,PRFLGS(R4)        ;SEEK NEED TO BE VERIFIED?
013024  001402                          BEQ     1$                       ;NO, CONTINUE
013026  000137  013650                  JMP     RDHFNC                   ;GO VERIFY SEEK
013032  005737  007532          1$:     TST     T.DRP                    ;DROP ON ERROR LIMITS REACHED?
013036  001456                          BEQ     8$                       ;NO
013040  026437  000012  007472          CMP     ERRCNT(R4),ERLMT         ;HARD REACHED?
013046  103404                          BLO     9$
013050  012737  003111  002116          MOV     #ERLMTM,WHY
013056  000442                          BR      11$
013060  026437  000014  007536  9$:     CMP     SFTCNT(R4),SFLMT         ;SOFT REACHED?
013066  103404                          BLO     10$
013070  012737  003154  002116          MOV     #SFEMSG,WHY
013076  000432                          BR      11$
013100  026437  000074  007564  10$:    CMP     DATCER(R4),T.DCD
013106  103404                          BLO     110$
013110  012737  003176  002116          MOV     #DCDMSG,WHY
013116  000422                          BR      11$
013120  016401  000016          110$:   MOV     SKECNT(R4),R1
013124  066401  000072                  ADD     TRERR(R4),R1
013130  020137  007474                  CMP     R1,SELMT
013134  103404                          BLO     7$
013136  012737  003133  002116          MOV     #SERLMT,WHY
013144  000407                          BR      11$
013146  026437  000020  007542  7$:     CMP     DERCNT(R4),DRLMT         ;DRIVE ERROR REACHED?
013154  103407                          BLO     8$
013156  012737  003221  002116          MOV     #DERMSG,WHY
013164  004537  020144          11$:    JSR     R5,DRDRV                 ;DROP THIS DRIVE!!!
013170  000137  012510                  JMP     MAIN                     ;GO GET ANOTHER




013174  005737  007540          8$:     TST     T.STA           ;DO WE WISH TO DROP ON OPR LIMITS
013200  001422                          BEQ     98$             ;NO

013202  026437  000000  007500          CMP     SKCNT(R4),SKLMT ;PAST THE SEEK LIMIT??
013210  103416                          BLO     98$             ;NO, THEN GO TEST
013212  016400  000060                  MOV     RXFR3(R4),R0    ;GET READ COUNT
013216  066400  000062                  ADD     WXFR3(R4),R0    ;ADD IN WRITE COUNT
013222  020037  007476                  CMP     R0,DALMT        ;LIMIT  REACHED??
013226  103407                          BLO     98$             ;NO, THEN GO TEST
013230  012737  003400  002116          MOV     #SOPLMT,WHY
013236  004537  020144                  JSR     R5,DRDRV        ;DROP THE DRIVE
013242  000137  012510                  JMP     MAIN            ;GO FOR ANOTHER DRIVE

013246  004537  021140          98$:    JSR     R5,RAND         ;GET FUNCTION, LEGAL FUNCTIONS
                                                                ;ARE:  1 (WRITE CHECK)
                                                                ;    :  2 (GET STATUS)
                                                                ;    :  3 (SEEK)
                                                                ;    :  4 (RD HEADER)
                                                                ;    :  5 (WRITE)
                                                                ;    :  6 (READ)
                                                                ;0 & 7 ARE NOT LEGIT

013252  013702  002126                  MOV     LONUM,R2        ;GET IT
013256  042702  177770                  BIC     #177770,R2      ;MASK TO 0-7
013262  001001                          BNE     6$              ;IF 0, MAKE 1
```

```
013264  005202                                  INC     R2
013266  022702  000007                  6$:     CMP     #7,R2           ;IS IT 7?
013272  001001                                  BNE     5$              ;IF 7, MAKE 6
013274  005302                                  DEC     R2
013276  006302                          5$:     ASL     R2              ;SHIFT LEFT (X2)
013300  000172  017414                          JMP     @LIST(R2)       ;GO TO FUNCTION ROUTINE


                                        .SBTTL  ROUTINE TO SETUP AND ISSUE GET STATUS

                                        ;WE GET HERE BY FALLING THRU "LIST" W!TH A RANDOM FUNCTION OF 2.

013304  012764  000004  000044  GSTFNC: MOV     #GSTAT,FUNC(R4) ;LOAD GET STATUS
013312  012764  000003  000040          MOV     #GSBIT,BDA(R4)  ;SET GSBIT IN COMMAND WORD
013320  000137  014002                  JMP     ISSUE           ;GO ISSUE FUNCTION

                                        .SBTTL  ROUTINE TO SETUP AND ISSUE SEEK FUNCTION

                                        ;WE GET HERE BY FALLING THRU "LIST" WITH A RANDOM FUNCTION OF 3.
                                        ;WE WILL CALL "RAND" FOR A NEW DISK ADDRESS TO SEEK
                                        ;TO.  ANY TRACK BUT LAST IS LEGAL.  WE WILL ALSO INCREMENT
                                        ;IT'S SEEK COUNT

013324  005764  000114          SKFNC:  TST     RSEEK(R4)  ;TRYING TO RECOVER
013330  001411                          BEQ     98$             ;NO, CONTINUE
013332  016401  000050                  MOV     LSTHDR(R4),R1   ;YES SET UP FOR RESEEK
013336  016402  000120                  MOV     PRPOS(R4),R2    ;TO CYLINDER
013342  042701  000100                  BIC     #100,R1         ;HEAD SET IN LATER
013346  042702  000100                  BIC     #100,R2         ;
013352  000507                          BR      4$              ;SKIP RANDOM PART
013354  004537  021140          98$:    JSR     R5,RAND         ;GET A RANDOM NUMBER
013360  013702  002126                  MOV     LONUM,R2        ;
013364  043702  002132                  BIC     SECMSK,R2       ;LEAVE CYL AND HEAD
013370  020264  000120                  CMP     R2,PRPOS(R4)    ;ON THAT TRACK ALREADY
013374  001002                          BNE     90$             ;NO, CONTINUE
```

```
013376  000137  013016                  JMP     GETFNC              ;YES, DON'T RESEEK
013402  005003                  90$:    CLR     R3
013404  010200                          MOV     R2,R0               ;COPY
013406  042700  177677                  BIC     #177677,R0          ;LEAVE ONLY HEAD
013412  023737  007520  007522          CMP     T.MXC,T.MNC         ;MIN AND MAX CYLINDERS THE SAME
013420  001003                          BNE     95$                 ;NO, BRANCH AND STAY IN LIMITS
013422  013702  007520                  MOV     T.MXC,R2            ;MAKE CYLINDER MAX/MIN
013426  000430                          BR      92$                 ;GO CLCULATE DIFF AND SEEK
013430  042702  000100          95$:    BIC     #HEAD,R2            ;STRIP OUT H.S. BIT
013434  023702  007520          94$:    CMP     T.MXC,R2            ;IS ADDRESS LESS/EQUAL THAN MAX
013440  103010                          BHIS    93$                 ;YES, CHECK LOW END
013442  005203                          INC     R3
013444  020327  000012                  CMP     R3,#10.
013450  001741                          BEQ     98$
013452  006202                          ASR     R2                  ;HALF IT AND CHECK AGAIN
013454  062702  000200          91$:    ADD     #BIT7,R2            ;JUST TO MAKE NON ZERO
013460  000763                          BR      95$                 ;GO BACK AND CHECK AGAIN
013462  023702  007522          93$:    CMP     T.MNC,R2            ;IS MIN GREATER/EQUAL THAN ADDRESS
013466  101410                          BLOS    92$                 ;YES, CALCULATE DIFF AND SEEK
013470  005203                          INC     R3
013472  020327  000012                  CMP     R3,#10
013476  001726                          BEQ     98$
013500  006302                          ASL     R2                  ;NO, DOUBLE IT
013502  042702  100000                  BIC     #BIT15,R2           ;BIT 15 CAN'T SET
013506  000762                          BR      91$                 ;GO CHECK MAX/MIN AGAIN
013510  016401  000120          92$.    MOV     PRPOS(R4),R1        ;GET PRESENT DISK POSITION
013514  043701  002130                  BIC     CYLMSK,R1           ;CLEAN OUT ITS SECTOR BITS

013520  016464  000120  000050          MOV     PRPOS(R4),LSTHDR(R4)     ;SAVE LAST
013526  010264  000120                  MOV     R2,PRPOS(R4)        ;NEW HEADER AFTER SEEK
013532  050064  000120                  BIS     R0,PRPOS(R4)        ;SET IN RANDOM HEAD GOTTEN
013536  023737  007514  007516          CMP     T.MXH,T.MNH         ;MIN AND MAX HEAD SELECT THE SAME
013544  001012                          BNE     96$                 ;NO, THEN WE CAN USE BOTH SURFACES
013546  005737  007514                  TST     T.MXH               ;WHICH IS OUR SURFACE FOR USE
013552  001004                          BNE     97$                 ;TOP SURFACE BRANCH
013554  042764  000100  000120          BIC     #HEAD,PRPOS(R4)     ;LOWER SURFACE ONLY
013562  000403                          BR      96$
013564  052764  000100  000120  97$:    BIS     #HEAD,PRPOS(R4)     ;TOP SURFACE ONLY
013572                          96$:
                                ;
                                ;CALCULATE THE DIFFERENCE WORD AND STORE IT IN BDA
                                ;

013572  160102                  4$:     SUB     R1,R2               ;SUBTRACT PRESENT FROM NEXT
013574  100002                          BPL     1$                  ;IF POSITIVE RESULT GO TO 1$
013576  005402                          NEG     R2                  ;NEG RESULT, NEGATE IT
013600  000402                          BR      2$                  ;GO SET DIRECTION OUT
013602  052702  000004          1$:     BIS     #SIGN,R2            ;DIRECTION OUT, MARKER
013606  052702  000001          2$      BIS     #MK,R2              ;MARKER BIT
013612  032764  000100  000120          BIT     #HEAD,PRPOS(R4)     ;WHICH SURFACE SELECTED?
013620  001402                          BEQ     3$                  ;TOP, THEN 3$
013622  052702  000020                  BIS     #SKHS,R2            ;BOTTOM SET HEAD BIT
013626  010264  000040          3$:     MOV     R2,BDA(R4)          ;MOVE DIFFERENCE WORD TO DA
013632  010264  000066                  MOV     R2,DIFWD(R4)        ;LOAD DIFFERENCE WORD
013636  012764  000006  000044          MOV     #SEEK,FUNC(R4)      ;LOAD SEEK
013644  000137  014002                  JMP     ISSUE
```

```

                                        .SBTTL  ROUTINE TO LOAD READ HEADER AND ISSUE IT.

                                        ;WE GET HERE BY FALLING THRU "LIST" WITH A RANDOM FUNCTION OF 4.
                                        ;

013650  012764  000010  000044  RDHFNC: MOV     #RDHDR,FUNC(R4) ;LOAD READ HEADER
013656  000137  014002                  JMP     ISSUE

                                        .SBTTL  ROUTINE TO LOAD WRITE DATA COMMAND


013662  022764  077700  000120  WRTFNC: CMP     #77700,PRPOS(R4)  ;ON LAST TRACK?
013670  001002                          BNE     98$             ;NO, CONTINUE
013672  000137  013324                  JMP     SKFNC           ;YES, WE'LL SEEK OFF IT!!
013676  005737  007544          98$:    TST     T.ROF           ;READ ONLY
013702  001402                          BEQ     97$             ;NO
013704  000137  013732                  JMP     RDDFNC          ;YES
013710  004537  022472          97$:    JSR     R5,GWCDA        ;GET WORD COUNT,DA
                                        ;
                                        ;WE NOW HAVE SECTOR AND WORD COUNT, LET'S WRITE BUFFER IN MEMORY
                                        ;TO WRITE OUT TO DISK
                                        ;FORMAT:        WORD 1 - # OF WORDS IN SECTOR
                                        ;       :       WORD 2 - ADDRESS OF PATTERN WRITTEN ON SECTOR
                                        ;       :       WORD 3 - 127  DATA PATTERN
                                        ;

013714  004537  017154                  JSR     R5,WRBUF        ;WRITE BUFFER INTO MEMORY
013720  012764  000012  000044          MOV     #WRITE,FUNC(R4) ;LOAD WRITE
013726  000137  014002                  JMP     ISSUE           ;GO ISSUE FUNCTION

                                        .SBTTL  ROUTINE TO LOAD READ DATA COMMAND

                                        ;THIS ROUTINE WILL FIRST CLEAR OUT THE BUFFER AREA,
                                        ;SELECT A RANDOM NUMBER OF WORDS TO READ AND A
                                        ;RANDOM SECTOR ON THE PRESENT CYLINDER TO READ FROM

013732  022764  077700  000120  RDDFNC: CMP     #77700,PRPOS(R4)  ;ON LAST TRACK?
013740  001002                          BNE     99$             ;NO CONTINUE
013742  000137  013324                  JMP     SKFNC           ;YES SEEK OFF IT.
013746  004537  022472          99$:    JSR     R5,GWCDA        ;GET WORD COUNT, DA
013752  016402  000042          97$:    MOV     BMP(R4),R2      ;CLEAR OUT BUFFER AREA
013756  017401  000110                  MOV     @BBA(R4),R1     ;SO WE KNOW READ
013762  005021                  1$:     CLR     (R1)+           ;WORKED!!
013764  005202                          INC     R2
013766  001375                          BNE     1$
013770  012764  000014  000044          MOV     #READ,FUNC(R4)  ;LOAD READ
013776  000137  014002                  JMP     ISSUE

                                        .SBTTL  SETUP CONTROLLER AND DRIVE INFO FOR INTERRUPT PROCESSING

                                        ;WE COME HERE BEFORE ISSUING ANY FUNCTION SO THAT ON INTERRUPT
                                        ;WE CAN PROPERLY PROCESS THE INTERRUPT.  WE WILL CHECK WHICH
                                        ;CONTROLLER WE ARE WORKING WITH AND STORE OFF THE DRIVE BUFFER
                                        ;POINTER IN IT'S "LSTDR"
                                        ;
```

```

014002  026437  000104  002134  ISSUE:  CMP     DCS(R4),CNTLR1   ;DRIVE ON CONTROLLER 1?
014010  001003                          BNE     1$               ;NO, ASSUME ON CONTROLLER 2
014012  010437  002140                  MOV     R4,LSTDR1        ;PUT BUFFER POINTER IN 1
014016  000402                          BR      2$               ;SKIP OVER NEXT INSTRUCTION
014020  010437  002142          1$:     MOV     R4,LSTDR2        ;PUT BUFFER POINTER IN 2
014024  052764  000100  000044  2$:     BIS     #INTEN,FUNC(R4)  ;ALLOW INTERRUPTS
014032  004537  014042                  JSR     R5,LDFUNC        ;NC WE ISSUE IT
014036  000137  012510                  JMP     MAIN             ;GO BACK AND DO ANOTHER


                                        .SBTTL  ROUTINE TO LOAD FUNCTION
                                        ;
                                        ;CALL   JSR     R5,LDFUNC
                                        ;ALL INFORMATION MUST BE SET UP IN DRIVE BUFFER
                                        ;R4 HAS POINTER TO BUFFER

014042  016403  000104          LDFUNC: MOV     DCS(R4),R3       ;GET CSR FOR DRIVE
014046  032713  000200                  BIT     #BIT7,(R3)       ;CAN WE ISSUE COMMAND?
014052  001003                          BNE     1$               ;YES, GO ISSUE COMMAND

014054                                  ERRSF   200.,PRGER       ;THIS ERROR SHOULD NEVER PRINT
   (3)  014054  104421                  TRAP    T$ERCODE
   (5)  014056  000310                  .WORD   200
   (5)  014060  002521                  .WORD   PRGER

014062  017463  000110  000002  1$:     MOV     @BBA(R4),BA(R3)  ;LOAD BUS ADDRESS REGISTER
014070  016463  000040  000004          MOV     BDA(R4),DA(R3)   ;LOAD DISK ADDRESS REGISTER
014076  016463  000042  000006          MOV     BMP(R4),MP(R3)   ;LOAD MULTI-PURPOSE REGISTER
014104  016464  000044  000046          MOV     FUNC(R4),BCSADR(R4)         ;GET FUNCTION
014112  056464  000106  000046          BIS     DRSEL(R4),BCSADR(R4)        ;SET DRIVE SELECT BITS
014120  052764  000201  000046          BIS     #CRDY!DRDY,BCSADR(R4)       ;SET CRDY: DRDY IN IMAGE
014126  042764  002000  000046          BIC     #OP1,BCSADR(R4)             ;WE'RE CLEAR BIT 10 FOR DRIVE 7-4 (OKAY?)
014134  016463  000046  000000          MOV     BCSADR(R4),CS(R3)           ;LOAD CSR
014142  042763  000200  000000          BIC     #CRDY,CS(R3)                ;ISSUE FUNCTION
014150  000205                          RTS     R5               ;EXIT

                                        .SBTTL  INTERRUPT SERVICE ROUTINES

014152                                  BGNSRV  INTR1


                                        ;ON INTERRUPT WE CHECK FOR ERRORS FIRST. IF NO ERRORS WE
                                        ;CHECK FUNCTION PREFORMED. WE ACT ACCORDING IF FUNCTION IS:
                                        ; 1-    WRITE CHECK - NOTHING IF NO ERROR
                                        ; 2-    GET STATUS - READ AND CHECK DRIVE STATUS
                                        ; 3-    SEEK - NOTHING RTI;  SET RD HDR AS NEXT COMMAND
                                        ; 4-    RDHDR - COMPARE HEADER TO PRESENT POSITION
                                        ; 5-    WRITE - UPDATE XFER COUNT, EXIT
                                        ; 6-    READ - COMPARE DATA IF REQUESTED, UPDATE XFER COUNT, EXIT
                                        ;
                                        ;ALL SUCCESSFUL EXITS FROM INTERRUPT ROUTINE TEST RETRY
                                        ;LIMIT IF RETRY IS LESS THEN LIMIT THEN LOG SOFT ERROR, CLEAR RETRY
                                        ;IF RETRY = 0, THEN NOTHING
                                        ;
                                        ;ON ERRORS - IF DRIVE ERROR - UNDER NON-INTERRUPT
                                        ;       DO:     GET STATUS - INVESTIGATE ERROR TYPE
                                        ;
                                        ;       DO:     DRIVE RESET - IF ERROR OCCURS AGAIN - FATAL ERROR
                                        ;                               IF NO ERROR, EXIT
                                        ;       DRIVE ERROR IS LOGGED UNDER ALL CIRCUMSTANCES
                                        ;
                                        ;
                                        ;       IF DCRC, HCRC, HNF CHECK BAD SECTOR LIST, IF IN LIST
                                        ;       IGNORE ERROR EXIT AS NORMAL, IF NOT IN LIST
                                        ;       INCREMENT RETRY; IF RETRY LIMIT EXCEEDED
                                        ;       LOG HARD ERROR, ELSE RETRY FUNCTION
                                        ;
                                        ;       IF OPI,NXM INCREMENT RETRY CHECK RETRY LIMIT
                                        ;       IF RETRY EXCEEDED LOG HARD ERROR EXIT
                                        ;       IF RETRY NOT EXCEEDED RETRY FUNCTION
                                        ;
                                        ;

014152  010446                          INTR1:  MOV     R4,-(SP)        ;SAVE PRESENT R4 VALUE
014154  013704  002140                          MOV     LSTDR1,R4       ;GET THE DRIVE BUFFER OF INTERRUPTING DRIVE
014160  000403                                  BR      SAVE            ;GO SAVE R0-R3
014162  010446                          INTR2:  MOV     R4,-(SP)        ;SAVE PRESENT R4 VALUE
014164  013704  002142                          MOV     LSTDR2,R4       ;GET THE DRIVE BUFFER OF INTERRUPTING DRIVE
014170  013746  002234                  SAVE:   MOV     E.CS,-(SP)
014174  013746  002236                          MOV     E.BA,-(SP)
014200  013746  002240                          MOV     E.DA,-(SP)
014204  013746  002242                          MOV     E.MP,-(SP)
014210  013746  002244                          MOV     E.MP1,-(SP)
014214  013746  002246                          MOV     E.MP2,-(SP)
014220  013746  002156                          MOV     CHKSEC,-(SP)
014224  013746  002154                          MOV     HDRFND,-(SP)
014230  013746  002164                          MOV     TEMP1,-(SP)
014234  013746  002116                          MOV     WHY,-(SP)
014240  013746  002310                          MOV     OPCALL,-(SP)
014244  013746  002312                          MOV     INCALL,-(SP)
014250  010346                                  MOV     R3,-(SP)        ;SAVE R3
014252  010246                                  MOV     R2,-(SP)        ;R2
014254  010146                                  MOV     R1,-(SP)        ;R1
014256  010046                                  MOV     R0,-(SP)        ;R0
014260  016403  000104                          MOV     DCS(R4),R3      ;GET CSR FOR INTERRUPT
014264  016337  000000  002234                  MOV     CS(R3),E.CS     ;SAVE ALL REGISTERS NOW!!
014272  016337  000002  002236                  MOV     BA(R3),E.BA
014300  016337  000004  002240                  MOV     DA(R3),E.DA
014306  016337  000006  002242                  MOV     MP(R3),E.MP
014314  016337  000006  002244                  MOV     MP(R3),E.MP1
014322  016337  000006  002246                  MOV     MP(R3),E.MP2
014330  005737  002234                          TST     E.CS            ;ANY ERRORS?
014334  100402                                  BMI     1$              ;YES, GO SOLVE ERROR MYSTERY
014336  000137  015456                          JMP     CHKFNC          ;NO, GO SEE IF WE HAVE TO DO ANYTHING

                                        .SBTTL  CONTROLLER ERROR CHECK ROUTINE

                                        ;WE HAVE SOME SORT OF ERROR LET'S FIND OUT WHICH ONE
                                        ;IT IS.

014342  013764  002240  000064          1$:     MOV     E.DA,LSTDA(R4)  ;SAVE DA FOR SOFT ERROR PRINT
```

```
      014350  032737  040000  002234          BIT     #DERR,E.CS          ;DRIVE ERROR?
      014356  001402                          BEQ     2$                  ;NO, CONTINUE
      014360  000137  016440                  JMP     CKDERR              ;YES, GO CHECK DRIVE ERROR
      014364  032737  000001  002234  2$:     BIT     #DRDY,E.CS          ;DRIVE READY THERE
      014372  001017                          BNE     23$                 ;YES, CONTINUE CHECKING
      014374  004537  021046                  JSR     R5,GETDST           ;NO,GET DRIVE STATUS
      014400  042701  000100                  BIC     #100,R1             ;GET RID OF HEAD
      014404  020127  000034                  CMP     R1,#34              ;ALLOW ONLY SEEK TRACKING STATE
      014410  001410                          BEQ     23$                 ;WAS 34 SKIP ERROR

      014412  005264  000012                  INC     ERRCNT(R4)          ;INDICATE HARD ERROR
      014416                                  ERRDF   1000.,NORDY,ERR9
 (3)  014416  104462                          TRAP    T$ERCODE
 (5)  014420  001750                          .WORD   1000
 (5)  014422  002473                          .WORD   NORDY
 (5)  014424  005034                          .WORD   ERR9

      014426  000137  016270                  JMP     EXIT1

      014432  032737  020000  002234  23$:    BIT     #NXM,E.CS           ;NON-EXISTANT MEMORY?
      014440  001407                          BEQ     3$                  ;NO, KEEP CHECKING
      014442  012764  004137  000052          MOV     #MTNXM,RTYPE(R4)    ;ERROR MESSAGE
      014450  005264  000034                  INC     NXMCNT(R4)          ;LOG ERROR
      014454  000137  015062                  JMP     111$                ;CHECK RETRY, EXIT BACK

      014460  032737  014000  002234  3$:     BIT     #BIT12!BIT11,E.CS   ;QUALIFING BITS SET?
      014466  001020                          BNE     5$                  ;YES, CAN'T BE OPI ALONE

      014470  032737  002000  002234          BIT     #OPI,E.CS           ;OPI SET?
      014476  001006                          BNE     4$                  ;YES, CONTINUE

      014500                                  ERRSF   10.,UDERR,ERR1      ;WE HAVE AN UNDIAGNOSABLE CONDITION, ONLY COMPOSITE SET
 (3)  014500  104461                          TRAP    T$ERCODE
 (5)  014502  000012                          .WORD   10
 (5)  014504  002576                          .WORD   UDERR
 (5)  014506  004270                          .WORD   ERR1
      014510                          33$:    BREAK
 (3)  014510  104022                          EMT     C$BRK
      014512  000776                          BR      33$


      014514  012764  004132  000052  4$:     MOV     #MTOPI,RTYPE(R4);SET UP FOR "OPI" PRINT
      014522  005264  000030                  INC     OPICNT(R4)          ;LOG ERROR
      014526  000555                          BR      111$                ;CHECK RETRY EXIT BACK

                                      ;WE KNOW IT'S NOW EITHER DLT, DCRC,HNF, OR HCRC
                                      ;CHECK FOR EACH

      014530  032737  002000  002234  5$:     BIT     #OPI,E.CS           ;OPI QUALIF!ER SET?
      014536  001060                          BNE     7$                  ;YES, THEN IT'S HCRC OR HNF

                                      ;IT'S NOW DOWN TO DLT OR DCRC

      014540  032737  010000  002234          BIT     #DLT,E.CS           ;DATA LATE?
      014546  001406                          BEQ     6$                  ;NO, MUST BE DATA CRC
      014550  012764  004125  000052          MOV     #MTDLT,RTYPE(R4);SET UP FOR "DLT"PRINT
```

```
     014556  005264  000026                    INC     DLTCNT(R4)        ;LOG ERROR
     014562  000537                            BR      111$              ;CHECK RETRY, EXIT

     014564  013737  002240  002156  6$:       MOV     E.DA,CHKSEC       ;SET UP SECTOR TO LOOK FOR
     014572  005364  000064                    DEC     LSTDA(R4)         ;DOWN COUNT FOR PRINT OUT
     014576  005337  002156                    DEC     CHKSEC            ;DOWN COUNT FOR LOOP UP
     014602  004537  023640                    JSR     R5,CKBDSC         ;CHECK BAD SECTOR LIST
     014606  005737  002154                    TST     HDRFND            ;WAS HEADER THERE?
     014612  001115                            BNE     110$              ;IGNORE ERROR, RETURN
     014614  005264  000022          117$:     INC     DCRCER(R4)        ;ACCOUNT FOR ERROR
     014620  012764  004120  000052            MOV     #MTDCRC,RTYPE(R4);SET UP FOR "DCRC" PRINT
     014626  022764  000102  000044            CMP     #INTEN!WRCHK,FUNC(R4)
     014634  001001                            BNE     118$
     014636  000511                            BR      111$

     014640  005737  007530          118$:     TST     T.DCK             ;DUMP BUFFER?
     014644  001506                            BEQ     111$              ;NO, EXIT
     014646                                    PRINTF  #FMT14,#DMPDCK
 (8) 014646  012746  003054                    MOV     #DMPDCK,-(SP)
 (7) 014652  012746  006473                    MOV     #FMT14,-(SP)
 (6) 014656  012746  000002                    MOV     #2,-(SP)
 (3) 014662  010600                            MOV     SP,R0
 (4) 014664  104017                            EMT     C$PNTF
 (4) 014666  062706  000006                    ADD     #6,SP
     014672  004537  023000                    JSR     R5,DMPBUF         ;DUMP BUFFER

     014676  000471                            BR      111$              ;EXIT

                                     ;IT'S NOW EITHER HNF OR HCRC.
                                     ;IF HCRC AND RDHDR, DETERMINE IF BAD SECTOR BY DOING 40 RDHDRS
                                     ;IF HCRC AND R/W, CHECK IF DA IS IN BAD SECTOR FILE
                                     ;IF HNF READ HEADER TO VERIFY IF ON CORRECT CYLINDER
                                     ;THEN IF ON CORRECT CYLINDER SEE IF DA IS A BAD SECTOR
                                     ;IF NOT ON CORRECT CYLINDER REPORT MISSEEK, LOG MISEEK
                                     ;AND PRESENT POSITION UPDATE.

     014700  032737  010000  002234  7$:       BIT     #HNF,E.CS         ;HEADER NOT FOUND SET?
     014706  001466                            BEQ     112$              ;NO IT MUST BE HCRC
     014710  012701  000051                    MOV     #41.,R1           ;ALLOW FOURTY READ HEADERS TO
     014714  004537  021062          8$:       JSR     R5,ISDRST
     014720  016402  000106                    MOV     DRSEL(R4),R2      ;FIND CYLINDER
     014724  052702  000010                    BIS     #RDHDR,R2         ;READ HEADER
     014730  016403  000104                    MOV     DCS(R4),R3
     014734  010263  000000                    MOV     R2,CS(R3)         ;ISSUE READ HEADER
     014740  004537  020776                    JSR     R5,WTRDY          ;WAIT
     014744  005301                            DEC     R1                ;DONE 40 OF THESE?
     014746  001422                            BEQ     9$                ;YES, GIVE UP WE DON'T HAVE ALL
     014750  005763  000000                    TST     CS(R3)            ;DAY, IS ERROR SET?
     014754  100757                            BMI     8$                ;YES, GO DO IT AGAIN

     014756  016301  000006                    MOV     MP(R3),R1         ;GET HEADER
     014762  043701  002132                    BIC     SECMSK,R1         ;MASK OUT SECTOR BITS
     014766  020164  000120                    CMP     R1,PRPOS(R4)      ;IS CYLINDER HEAD CORRECT?
     014772  001415                            BEQ     10$               ;YES, GO CHECK BAD SECTOR LIST


```

```
      014774  005264  000072                   INC     TRERR(R4)
      015000                                   ERRHRD  20.,TRACK,ERR2  ;TRACKING DRIFT ERROR
 (3)  015000  104463                           TRAP    T$ERCODE
 (5)  015002  000024                           .WORD   20
 (5)  015004  003074                           .WORD   TRACK
 (5)  015006  004276                           .WORD   ERR2


      015010  000137  015760                   JMP     SKRETRY         ;FIX TRACKING ERROR


      015014                           9$:     ERRHRD  30.,EXHAUS,ERR1 ;WE CAN'T FIND GOOD HEADER ON THIS TRACK
 (3)  015014  104463                           TRAP    T$ERCODE
 (5)  015016  000036                           .WORD   30
 (5)  015020  002562                           .WORD   EXHAUS
 (5)  015022  004270                           .WORD   ERR1

      015024  000410                           BR      110$

      015026  013737  002240  002156   10$:    MOV     E.DA,CHKSEC
      015034  004537  023640                   JSR     R5,CKBDSC       ;GO CHECK BAD SECTOR FILE
      015040  005737  002154                   TST     HDRFND          ;WAS IT THERE
      015044  001401                           BEQ     11$             ;NO, LOG IT EXIT
      015046  000577                   110$:   BR      GOERRX          ;YES IGNORE ERROR

      015050  005264  000032           11$.    INC     HNFERR(R4)      ;LOG IT
      015054  012764  004105  000052           MOV     #MTHNF,RTYPE(R4);SET UP FOR "HNF" PRINT
      015062  000573                   111$:   BR      GOFIN           ;EXIT



                                       ;
                                       ;IT WAS A HEADER CRC ERROR, FIGURE OUT IF IT WAS
                                       ;ON A READ HEADER OR READ/WRITE
                                       ;

      015064  022764  000110  000044   112$:   CMP     #INTEN!RDHDR,FUNC(R4)   ;READ HEADER?
      015072  001417                           BEQ     13$             ;YES, GO FIND OUT MORE ABOUT IT
                                                                       ;NO, IT MUST BE R/W
      015074  013737  002240  002156           MOV     E.DA,CHKSEC
      015102  004537  023640                   JSR     R5,CKBDSC       ;BAD SECTOR SEARCH
      015106  005737  002154                   TST     HDRFND          ;WAS OUR DA THERE?
      015112  001401                           BEQ     12$             ;NO, MUST BE LEGIT ERROR
      015114  000554                           BR      GOERRX          ;YES, IGNORE ERROR

      015116  005264  000024           12$:    INC     HCRCER(R4)      ;LOG ERROR
      015122  012764  004112  000052           MOV     #MTHCRC,RTYPE(R4)
      015130  000550                           BR      GOFIN

      015132  017401  000110           13$:    MOV     @BBA(R4),R1     ;USE IT'S BUFFER TO STORE HDRS
      015136  012737  000050  002164           MOV     #40.,TEMP1      ;40 CONSECUTIVE HEADERS
      015144  012702  000010           14$:    MOV     #RDHDR,R2       ;READ HEADER
      015150  056402  000106                   BIS     DRSEL(R4),R2    ;
      015154  016403  000104                   MOV     DCS(R4),R3      ;
      015160  010263  000000                   MOV     R2,CS(R3)       ;
      015164  004537  020776                   JSR     R5,WTRDY        ;WAIT FOR READY
      015170  016321  000000                   MOV     CS(R3),(R1)+    ;READ ALL REGISTERS
```

```
015174  016321  000006                    MOV     MP(R3),(R1)+     ;
015200  016321  000006                    MOV     MP(R3),(R1)+     ;
015204  016321  000006                    MOV     MP(R3),(R1)+     ;
015210  005337  002164                    DEC     TEMP1            ;DONE 40 YET?
015214  001353                            BNE     14$              ;NO, GO BACK

                                  ;WE HAVE 40 HEADERS NOW LETS SEE IF WE CAN VERIFY WHETHER
                                  ;OR NOT A BAD SECTOR CAUSED THE ERROR.  CHECK FIRST TO SEE
                                  ;IF WE HAVE ANY BAD SECTORS ON THIS TRACK.

015216  017402  000110            99$:    MOV     @BBA(R4),R2      ;GET BUFFER START
015222  012701  000050                    MOV     #40.,R1          ;FOURTY HEADERS
015226  032712  002000            15$:    BIT     #OPI,(R2)        ;IS OPI SET IN CS
015232  001403                            BEQ     16$              ;NO, WELL CAN'T BE HCRC
015234  032712  004000                    BIT     #HCRC,(R2)       ;INSURE HCRC W/OPI
015240  001005                            BNE     17$              ;FOUND GO SEE IF IT COMPARES
015242  062702  000010            16$:    ADD     #10,R2           ;NEXT CS IMAGE
015246  005301                            DEC     R1               ;DONE 40
015250  001366                            BNE     15$
015252  000721                            BR      12$

015254  020274  000110            17$:    CMP     R2,@BBA(R4)      ;IS HEADER FIRST ONE?
015260  001046                            BNE     21$              ;NO, READ PREVIOUS HEADER
                                                                   ;YES, WE'LL HAVE TO GO THRU
                                                                   ;AND CHECK OTHERS BEFORE WE
                                                                   ;CAN SAFELY CALCULATE
                                                                   ;"SUPPOSED" BAD SECTOR
015262  017401  000110                    MOV     @BBA(R4),R1
015266  012703  000001                    MOV     #1,R3
015272  062701  000010            18$:    ADD     #10,R1
015276  032711  002000                    BIT     #OPI,(R1)
015302  001416                            BEQ     19$
015304  032711  004000                    BIT     #HCRC,(R1)
015310  001413                            BEQ     19$
015312  005203                            INC     R3
015314  022703  000017                    CMP     #15.,R3
015320  001364                            BNE     18$


015322  012737  003457  002116            MOV     #MBDMSC,WHY      ;DROP DRIVE DUE TO
015330  004537  020144                    JSR     R5,DRDRV         ;MORE THAN 16 BAD SECTORS
015334  000137  016270                    JMP     EXIT1


015340  005012                    19$:    CLR     (R2)             ;CLEAR THIS CS
015342  062701  000002                    ADD     #2,R1            ;GET IT'S HEADER ADDRESS
015346  011102                            MOV     (R1),R2          ;GET HEADER
015350  010201                            MOV     R2,R1            ;SAVE HEADER
015352  042702  177700                    BIC     #177700,R2       ;MASK   ONLY SECTOR
015356  160301                            SUB     R3,R1            ;BACK UP TO SECTOR WHICH IS BAD
015360  100402                            BMI     20$              ;IF MINUS DO MAGIC
015362  160302                            SUB     R3,R2            ;NO THEN SUBTRACT IS LEGAL
015364  000421                            BR      22$              ;BRANCH TO CHECK FILE
015366  160302                    20$:    SUB     R3,R2            ;THIS SUB PRODUCES WRONG ANSWER
015370  062702  000050                    ADD     #50,R2           ;FIX IT UP
015374  000415                            BR      22$              ;GO CHECK FILE
```

```

015376  005012                  21$:    CLR     (R2)                    ;CLEAR THIS CS OUT
015400  162702  000006                  SUB     #6,R2                   ;GET PREVIOUS HEADER
015404  011201                          MOV     (R2),   R1
015406  005201                          INC     R1
015410  010102                          MOV     R1,R2
015412  042701  177700                  BIC     #177700,R1
015416  022701  000050                  CMP     #40.,R1
015422  002402                          BLT     22$
015424  162702  000050                  SUB     #40.,R2
015430  010237  002156          22$:    MOV     R2,CHKSEC
015434  004537  023640                  JSR     R5,CKBDSC
015440  005737  002154                  TST     HDRFND
015444  001664                          BEQ     99$
015446  000137  016274          GOERRX: JMP     ERREX


015452  000137  016376          GOFIN:  JMP     FINERR






                                .SBTTL  COMMAND SERVICE ROUTINES

                                ;THERE WAS NO ERROR SO............
                                ;NOW WE WILL FIND OUT WHICH FUNCTION WE DID TO CAUSE
                                ;INTERRUPT AND ACT ACCORDINGLY.
                                ;

015456  016401  000044          CHKFNC: MOV     FUNC(R4),R1     ;GET FUNCTION OF DRIVE
015462  006201                          ASR     R1              ;ALIGN THE FUNCTION CODE
015464  042701  000040                  BIC     #40,R1          ;WIPE OUT INT. ENAB (SHIFTED)
015470  005301                          DEC     R1              ;WRITE CHECK??
015472  001004                          BNE     2$              ;NO, BRANCH
015474  004537  011366                  JSR     R5,CLRWCK
015500  000137  016236                  JMP     EXIT
015504  005301                  2$:     DEC     R1              ;GET STATUS?
015506  001555                          BEQ     AGSTAT          ;BRANCH IF SO
015510  005301                          DEC     R1              ;SEEK?
015512  001416                          BEQ     ASEEK           ;BRANCH IF SO
015514  005301                          DEC     R1              ;RDHDR?
015516  001470                          BEQ     ARDHDR          ;BRANCH IF SO
015520  005301                          DEC     R1              ;WRITE?
015522  001002                          BNE     1$              ;NO, BRANCH
015524  000137  016146                  JMP     AWRITE
015530  005301                  1$:     DEC     R1              ;READ?
015532  001425                          BEQ     AFREAD          ;BRANCH IF SO

015534                                  ERRSF   210.,PRGER
(3) 015534  104421                      TRAP    T$ERCODE
(5) 015536  000322                      .WORD   210
(5) 015540  002521                      .WORD   PRGER

015542  000000                          HALT
015544  000137  016236          XEXIT:  JMP     EXIT
```

```
  4487
  4488                                       .SBTTL                  SEEK
  4489
  4490  015550  052764  000001  000056  ASEEK:  BIS     #SKDON,PRFLGS(R4)  ;SET SEEK VERIFY NEEDED
  4491  015556  005264  000054                  INC     SKCNT1(R4)         ;INCREMENT COUNT
  4492  015562  026427  000054  001750          CMP     SKCNT1(R4),#1000.  ;10(3) REACHED
  4493  015570  002404                          BLT     99$                ;NO, EXIT
  4494  015572  005264  000000                  INC     SKCNT(R4)          ;YES, BUMP THOUSANDS
  4495  015576  005064  000054                  CLR     SKCNT1(R4)
  4496  015602  000137  016274          99$:    JMP     ERREX
  4497
  4498                                       .SBTTL                  READ
  4499
  4500  015606                          AFREAD: SETPRI  #340
   (3)  015606  012700  000340                  MOV     #340,R0
   (3)  015612  104041                          EMT     C$SPRI
  4501  015614  004537  020424                  JSR     R5,CKDATA          ;CHECK DATA
  4502  015620                          AFWRCK:
  4503  015620  016401  000042          1$:     MOV     BMP(R4),R1         ;BUMP UP XFER COUNT
  4504  015624  005401                          NEG     R1                 ;MAKE POSITiVE
  4505  015626  060164  000002                  ADD     R1,RXFR1(R4)       ;ADD THE BITS
  4506  015632  022764  023420  000002          CMP     #10000.,RXFR1(R4)  ;10(8) REACHED YET
  4507  015640  101016                          BHI     2$                 ;NO, EXIT
  4508  015642  005264  000004                  INC     RXFR2(R4)          ;BUMP 10(10)
  4509  015646  162764  023420  000002          SUB     #10000.,RXFR1(R4)  ;START 10(8) AT 0
  4510  015654  022764  023420  000004          CMP     #10000.,RXFR2(R4)  ;10(10) REACHED YET
  4511  015662  101005                          BHI     2$                 ;NO, EXIT
  4512  015664  005264  000060                  INC     RXFR3(R4)          ;YES BUMP 65K 10(10)
  4513  015670  162764  023420  000004          SUB     #10000.,RXFR2(R4)  ;MAKE 10(10) 0
  4514  015676  000557                  2$:     BR      EXIT               ;EXIT
  4515
  4516                                       .SBTTL                  READ HEADER
  4517
  4518  015700  013701  002242          ARDHDR: MOV     E.MP,R1            ;GET HEADER
  4519  015704  043701  002132                  BIC     SECMSK,R1          ;MASK OUT SECTOR BITS
  4520  015710  026401  000120                  CMP     PRPOS(R4),R1       ;IS HEADER CORRECT?
  4521  015714  001442                          BEQ     1$                 ;YES, CONTINUE
  4522
  4523  015716  032764  000001  000056          BIT     #SKDON,PRFLGS(R4)  ;IS THIS MIS-SEEK OR TRACKING ERROR
  4524  015724  001407                          BEQ     2$                 ;BRANCH IF TRACKING
  4525
  4526  015726  005264  000016                  INC     SKECNT(R4)         ;ACCOUNT FOR SEEK ERROR
  4527  015732                                  ERRHRD  50.,MSKER,ERR2
   (3)  015732  104463                          TRAP    T$ERCODE
   (5)  015734  000062                          .WORD   50
   (5)  015736  002620                          .WORD   MSKER
   (5)  015740  004276                          .WORD   ERR2
  4528  015742  000406                          BR      3$                 ;BRANCH AROUND TRACKING ERROR REPORT
  4529
  4530  015744  005264  000072          2$:     INC     TRERR(R4)          ;ACCOUNT FOR TRACKING ERROR
  4531  015750                                  ERRHRD  55.,TRACK,ERR2     ;TRACKING ERROR
   (3)  015750  104463                          TRAP    T$ERCODE
   (5)  015752  000067                          .WORD   55
   (5)  015754  003074                          .WORD   TRACK
   (5)  015756  004276                          .WORD   ERR2
  4532
```

```
                 015760                                SKRETRY=.

         015760  005264  000114              3$:       INC     RSEEK(R4)  ;SET RETRY IN PROGRESS
         015764  026437  000114  007552                CMP     RSEEK(R4),T.SLT ;RETRY EXHAUSTED?????
         015772  101405                                BLOS    4$             ;NO, THEN RETRY

         015774                                        ERRHRD  333.,SEXHAU,ERR2
    (3)  015774  104463                                TRAP    T$ERCODE
    (5)  015776  000515                                .WORD   333
    (5)  016000  003312                                .WORD   SEXHAU
    (5)  016002  004276                                .WORD   ERR2
         016004  000406                                BR      1$

         016006  010164  000050              4$:       MOV     R1,LSTHDR(R4)   ;SET UP RETRY
         016012  042764  000001  000056                BIC     #SKDON,PRFLGS(R4)  ;ALLOW SEEK
         016020  000506                                BR      EXIT            ;EXIT
         016022  042764  000001  000056      1$:       BIC     #SKDON,PRFLGS(R4)  ;SET VERIFICATION DONE
         016030  005064  000114                        CLR     RSEEK(R4)
         016034  010164  000120                        MOV     R1,PRPOS(R4)    ;MAKE THIS HEADER PRESENT POSITION
         016040  000476                                BR      EXIT            ;EXIT

                                              SBTTL .             GET STATUS

         016042  013701  002242              AGSTAT:  MOV     E.MP,R1         ;GET STATUS
         016046  042701  000100                        BIC     #100,R1         ;CLEAR OUT HEAD SELECT
         016052  005737  007544                        TST     T.ROF           ;READ ONLY
         016056  001402                                BEQ     2$
         016060  042701  020000                        BIC     #WL,R1
         016064  032701  177400              2$:       BIT     #177400,R1      ;ANY BITS WRONG
         016070  001406                                BEQ     1$              ;NO. CONTINUE

         016072  005264  000012                        INC     ERRCNT(R4)      ;STATUS BITS WRONG
         016076                                        ERRHRD  60.,MDSER,ERR4
    (3)  016076  104463                                TRAP    T$ERCODE
    (5)  016100  000074                                .WORD   60
    (5)  016102  002705                                .WORD   MDSER
    (5)  016104  004500                                .WORD   ERR4

         016106  010102                      1$:       MOV     R1,R2           ;COPY STATUS WORD
         016110  042702  177700                        BIC     #177700,R2      ;GET STATE BITS
         016114  022702  000034                        CMP     #34,R2          ;COVER CLSD, HEADS OUT, BRUSHES HOME, SEEK TRACK COUNTIN
         016120  001446                                BEQ     EXIT            ;YES, EXIT
         016122  022702  000035                        CMP     #35,R2          ;COVER CLSD, HEADS OUT, BRUSHES HOME, SEEK LINEAR MODE
         016126  001443                                BEQ     EXIT            ;YES, EXIT

         016130  005264  000012                        INC     ERRCNT(R4)
         016134                                        ERRHRD  70.,MDSER,ERR4
    (3)  016134  104463                                TRAP    T$ERCODE
    (5)  016136  000106                                .WORD   70
    (5)  016140  002705                                .WORD   MDSER
    (5)  016142  004500                                .WORD   ERR4

         016144  000434                                BR      EXIT

                                              SBTTL               WRITE

```

```
 4577  016146  016401  000042          AWRITE: MOV     BMP(R4),R1         ;GET WORD COUNT
 4578  016152  005401                          NEG     R1                 ;MAKE POSITIVE
 4579  016154  060164  000006                  ADD     R1,WXFR1(R4)       ;ADD THE BITS
 4580  016160  022764  023420  000006          CMP     #10000.,WXFR1(R4)  ;10(5) YET?
 4581  016166  101023                          BHI     EXIT               ;NO, EXIT
 4582  016170  005264  000010                  INC     WXFR2(R4)          ;YES BUMP 10(10)
 4583  016174  162764  023420  000006          SUB     #10000.,WXFR1(R4)  ;10(5) GOES TO ZERO
 4584  016202  022764  023420  000010          CMP     #10000.,WXFR2(R4)  ;10(10) YET?
 4585  016210  101012                          BHI     EXIT               ;NO EXIT
 4586  016212  005264  000062                  INC     WXFR3(R4)          ;INC 65K (10)(10)
 4587  016216  162764  023420  000010          SUB     #10000.,WXFR2(R4)  ;MAKE 10(10)
 4588  016224  005737  007562                  TST     T.WCK              ;PERFORM WRITE CHECK
 4589  016230  001402                          BEQ     EXIT
 4590  016232  004537  011330                  JSR     R5,SETWCK
 4591
 4592  016236  005764  000036          EXIT:   TST     RETRY(R4)          ;IN PROCESS OF RETRYING?
 4593  016242  001414                          BEQ     ERREX              ;NO
 4594  016244  026427  000052  004144          CMP     RTYPE(R4),#MTDRV
 4595  016252  001406                          BEQ     EXIT1
 4596  016254  005264  000014                  INC     SFTCNT(R4)         ;YES, LOG SOFT ERROR
 4597
 4598  016260                                  ERRSOFT 80.,MSFER,ERR3     ;REPORT SOFT ERROR
  (3)  016260  104464                          TRAP    TSERCODE
  (5)  016262  000120                          .WORD   80
  (5)  016264  002631                          .WORD   MSFER
  (5)  016266  004362                          .WORD   ERR3
 4599
 4600  016270  005064  000036          EXIT1:  CLR     RETRY(R4)          ;CLEAR RETRY
 4601
 4602  016274  042774  000100  000104  ERREX:  BIC     #INTEN,@DCS(R4)
 4603  016302  012600                          MOV     (SP)+,R0
 4604  016304  012601                          MOV     (SP)+,R1
 4605  016306  012602                          MOV     (SP)+,R2
 4606  016310  012603                          MOV     (SP)+,R3
 4607  016312  012637  002312                  MOV     (SP)+,INCALL
 4608  016316  012637  002310                  MOV     (SP)+,OPCALL
 4609  016322  012637  002116                  MOV     (SP)+,WHY
 4610  016326  012637  002164                  MOV     (SP)+,TEMP1
 4611  016332  012637  002154                  MOV     (SP)+,HDRFND
 4612  016336  012637  002156                  MOV     (SP)+,CHKSEC
 4613  016342  012637  002246                  MOV     (SP)+,E.MP2
 4614  016346  012637  002244                  MOV     (SP)+,E.MP1
 4615  016352  012637  002242                  MOV     (SP)+,E.MP
 4616  016356  012637  002240                  MOV     (SP)+,E.DA
 4617  016362  012637  002236                  MOV     (SP)+,E.BA
 4618  016366  012637  002234                  MOV     (SP)+,E.CS
 4619  016372  012604                          MOV     (SP)+,R4
 4620  016374                                  ENDSRV
  (3)  016374                          L10023:
  (2)  016374  000002                          RTI
 4621
 4622  016376  004537  017374          FINERR: JSR     R5,RCNT            ;CHECK TO SEE IF WE HAVE EXCEEDED
 4623  016402  000405                          BR      1$                 ;RETRY LIMIT. IF SO 1$ AND REPORT HARD
 4624  016404  013764  002234  000116          MOV     E.CS,SOFTCS(R4)
 4625  016412  000137  016274                  JMP     ERREX              ;NOT EXCEEDED EXIT
 4626  016416  005264  000012          1$:     INC     ERRCNT(R4)         ;INDICATE ERROR
 4627
 4628  016422                               ERRHRD  90.,MHDER,ERR1  ;NON-RECOVERABLE ERROR
  (3)  016422  104463                       TRAP    T$ERCODE
  (5)  016424  000132                       .WORD   90
  (5)  016426  003041                       .WORD   MHDER
  (5)  016430  004270                       .WORD   ERR1
 4629  016432  004537  011366               JSR     R5,CLRWCK
 4630
 4631  016436  000714                       BR      EXIT1                     :
 4632
 4633                               .SBTTL  DRIVE ERROR SERVICE
 4634
 4635                               ;WE HAVE A DRIVE ERROR, LET'S GET THE STATUS
 4636
 4637  016440  005264  000020       CKDERR: INC     DERCNT(R4)        ;ACCOUNT FOR ERROR
 4638  016444  004537  021046               JSR     R5,GETDST         ;GET DRIVE STATUS
 4639                               ;REPOFT DRIVE ERROR
 4640  016450                               ERRHRD  224.,DRVER,ERR9 ;DRIVE ERROR
  (3)  016450  104463                       TRAP    T$ERCODE
  (5)  016452  000340                       .WORD   224
  (5)  016454  002650                       .WORD   DRVER
  (5)  016456  005034                       .WORD   ERR9
 4641
 4642                               ;ACT ACCORDINGLY TO DRIVE ERROR
 4643
 4644  016460  032701  001000               BIT     #VC,R1            ;VOLUME CHECK?
 4645  016464  001027                       BNE     9$                ;YES, GO ISSUE RESET
 4646  016466  032701  010000               BIT     #SKTO,R1          ;SEEK TIME OUT?
 4647  016472  001070                       BNE     12$               ;YES, ISSUE RESET
 4648  016474  032701  144000               BIT     #WDE!HCE!SPE,R1 ;WRITE DATA, CURRENT HEAD, SPINDLE?
 4649  016500  001130                       BNE     15$               ;GO WAIT FOR HEADS TO UNLOAD
 4650  016502  032701  002000               BIT     #WGE,R1           ;WRITE GATE ERROR
 4651  016506  001003                       BNE     20$               ;YES, ISSUE  RESET
 4652  016510  004537  021062               JSR     R5,ISDRST         ;ISSUE RESET
 4653  016514  000431                       BR      10$               ;GO CHECK DRIVE READY
 4654  016516  004537  021062       20$:    JSR     R5,ISDRST         ;ISSUE RESET
 4655  016522  004537  021046               JSR     R5,GETDST         ;RESET WORK?
 4656  016526  032701  002000               BIT     #WGE,R1           ;WGE CLEAR
 4657  016532  001422                       BEQ     10$               ;YES GO CHECK DRIVE READY
 4658  016534  012737  002746  002116       MOV     #WGEST,WHY        ;REPOPT WGE DIDN'T CLR
 4659  016542  000412                       BR      91$               ;DROP DRIVE
 4660
 4661  016544  004537  021062       9$:     JSR     R5,ISDRST         ;ISSUE RESET
 4662  016550  004537  021046               JSR     R5,GETDST         ;RESET WORK
 4663  016554  032701  001000               BIT     #VC,R1            ;VOL CHK CLEAR
 4664  016560  001407                       BEQ     10$               ;YES, CHECK DRIVE READY
 4665  016562  012737  002721  002116       MOV     #MVCER,WHY        ;DROP THE DRIVE
 4666
 4667  016570  004537  020144       91$:    JSR     R5,DRDRV
 4668  016574  000137  016270               JMP     EXIT1
 4669  016600  032763  000001  000000  10$:    BIT     #DRDY,CS(R3)      ;DRIVE READY POSTED?
 4670  016606  001004                       BNE     101$              ;YES, PRINT RECOVERED
 4671
 4672  016610  012737  002460  002116       MOV     #DNRDY,WHY
 4673  016616  000764                       BR      91$               ;NO. DROP DRIVE
 4674
```

```
4675  016620                           101$:   PRINTB  #FMT14,#MRDER     ;PRINT DRIVE RECOVERED
 (8)  016620  012746  003001                   MOV     #MRDER,-(SP)
 (7)  016624  012746  006473                   MOV     #FMT14,-(SP)
 (6)  016630  012746  000002                   MOV     #2,-(SP)
 (3)  016634  010600                           MOV     SP,R0
 (4)  016636  104014                           EMT     C$PNTB
 (4)  016640  062706  000006                   ADD     #6,SP
4676  016644  004537  017102                   JSR     R5,GHDR
4677  016650  000137  016376                   JMP     FINERR
4678  016654  012702  000004           12$:    MOV     #4,R2             ;SEEK TIME OUT
4679  016660  004537  021062           13$:    JSR     R5,ISDRST         ;ISSUE DRIVE RESET
4680                                                                     ;FOUR TIMES BEFORE
4681  016664                                   WAITUS  #15000.           ;DROPPING DRIVE
 (3)  016664  012700  035230                   MOV     #15000.,R0
 (3)  016670  104027                           EMT     C$WTU
4682
4683  016672  032763  000001  000000           BIT     #DRDY,CS(R3)      ;DRIVE READY YET?
4684  016700  001006                           BNE     14$               ;YES, CHECK IF ERROR CLEARED
4685  016702  005302                           DEC     R2                ;NO, HAVE WE DONE IT FOUR TIMES
4686  016704  001365                           BNE     13$               ;YET
4687
4688  016706  012737  002657  002116  141$:    MOV     #MDERS,WHY        ;YES, DROP DRIVE
4689  016714  000725                           BR      91$
4690
4691  016716  032763  040000  000000  14$:     BIT     #DEPR,CS(R3)      ;DRIVE ERROR SET STILL
4692  016724  001370                           BNE     141$              ;YES, DROP DRIVE
4693  016726                                   PRINTB  #FMT14,#MRDER
 (8)  016726  012746  003001                   MOV     #MRDER,-(SP)
 (7)  016732  012746  006473                   MOV     #FMT14,-(SP)
 (6)  016736  012746  000002                   MOV     #2,-(SP)
 (3)  016742  010600                           MOV     SP,R0
 (4)  016744  104014                           EMT     C$PNTB
 (4)  016746  062706  000006                   ADD     #6,SP
4694  016752  004537  017102                   JSR     R5,GHDR
4695  016756  000137  016236                   JMP     EXIT
4696
4697  016762  012702  000004           15$:    MOV     #4,R2             ;WAIT FOR HEADS TO UNLOAD
4698  016766  004537  021046           16$:    JSR     R5,GETDST         ;GET STATUS
4699  016772  032701  000020                   BIT     #BIT4,R1          ;UNLOAD STATE
4700  016776  001411                           BEQ     17$               ;YES, CONTINUE W/ RECOVERY
4701  017000                                   WAITMS  #1.               ;WAIT A WHILE
 (3)  017000  012700  000001                   MOV     #1.,R0
 (3)  017004  104026                           EMT     C$WTM
4702  017006  005302                           DEC     R2                ;WAIT LONG ENOUGH
4703  017010  001366                           BNE     16$               ;NO, GO BACK
4704  017012  012737  003336  002116           MOV     #UNLOAD,WHY       ;DROP DRIVE
4705  017020  000663                           BR      91$
4706
4707  017022  004537  021062           17$:    JSR     R5,ISDRST         ;ISSUE RESET
4708  017026                                   WAITMS  #1.               ;
 (3)  017026  012700  000001                   MOV     #1.,R0
 (3)  017032  104026                           EMT     C$WTM
4709  017034  032763  040000  000000           BIT     #DERR,CS(R3)      ;DRIVE ERROR CLEAR?
4710  017042  001321                           BNE     141$              ;NO, DROP DRIVE
4711  017044  012702  000075                   MOV     #61.,R2           ;YES, WAIT 60 SECONDS
4712  017050                           18$:    WAITMS  #10.              ;FOR DRIVE READY TO
```

```
   (3)  017050  012700  000012                    MOV     #10.,R0
   (3)  017054  104026                            EMT     C$WTM
  4713  017056  032763  000001  000000            BIT     #DRDY,CS(R3)      ;COME BACK
  4714  017064  001314                            BNE     14$               ;
  4715  017066  005302                            DEC     R2
  4716  017070  001367                            BNE     18$
  4717  017072  012737  003362  002116            MOV     #NOLOAD,WHY       ;NO READY DROP DRIVE
  4718  017100  000633                            BR      91$
  4719
  4720
  4721  017102  012763  000210  000000  GHDR:     MOV     #CRDY!RDHDR,CS(R3)
  4722  017110  056463  000106  000000            BIS     DRSEL(R4),CS(R3)
  4723  017116  042763  000200  000000            BIC     #200,CS(R3)
  4724  017124  004537  020776                    JSR     R5,WTRDY
  4725  017130  016301  000006                    MOV     MP(R3),R1
  4726  017134  043701  002132                    BIC     SECMSK,R1
  4727  017140  010164  000120                    MOV     R1,PRPOS(R4)
  4728  017144  012764  004144  000052            MOV     #MTDRV,RTYPE(R4)  ;SETUP DRIVE ERROR
  4729  017152  000205                            RTS     R5
  4730
  4731                                  ;ROUTINE TO WRITE A BUFFER INTO MEMORY.  USES WORD COUNT AND BUS
  4732                                  ;ADDRESS FROM DRIVE BUFFER (R4).  WILL WRITE RANDOM FROM ONE OF
  4733                                  ;8 PATTERNS.  USED BY WRITE FUNCTION AND WRPACK ROUTINE.
  4734
  4735  017154  010346                  WRBUF:    MOV     R3,-(SP)          ;SAVE REGISTERS
  4736  017156  010246                            MOV     R2,-(SP)
  4737  017160  010146                            MOV     R1,-(SP)
  4738  017162  010046                            MOV     R0,-(SP)
  4739  017164  016402  000042                    MOV     BMP(R4),R2        ;R2 HAS TOTAL WORDS TO SET UP FOR
  4740  017170  005402                            NEG     R2                ;POSITIVE NUMBER
  4741  017172  017401  000110                    MOV     @BBA(R4),R1       ;WHERE BUFFER IS
  4742  017176  020227  000200          2$:       CMP     R2,#128.          ;MORE THAN 128 WORDS
  4743  017202  002015                            BGE     4$                ;YES, BRANCH
  4744  017204  020227  000003                    CMP     R2,#3             ;GREATER THAN THREE WORDS
  4745  017210  002005                            BGE     3$                ;YES, BRANCH
  4746  017212  062702  000003                    ADD     #3,R2             ;ADD 3
  4747  017216  162764  000003  000042            SUB     #3,BMP(R4)        ;WC UP BY 3
  4748  017224  010221                  3$:       MOV     R2,(R1)+          ;STORE WC
  4749  017226  005302                            DEC     R2                ;ACCOUNT FOR WC
  4750  017230  010237  002176                    MOV     R2,TEMP6          ;LOAD DOWN COUNTER
  4751  017234  000405                            BR      5$
  4752  017236  012737  000177  002176  4$:       MOV     #127.,TEMP6       ;LOAD DOWN COUNTER
  4753  017244  012721  000200                    MOV     #128.,(R1)+
  4754  017250  005737  007546          5$:       TST     T.RAN             ;RANDOM SELECT OF PATTERNS
  4755  017254  001003                            BNE     55$               ;YEA
  4756  017256  013703  007550                    MOV     T.PAT,R3          ;NO GET PATTERN OPERATOR
  4757  017262  000406                            BR      56$               ;WANTS TO USE
  4758  017264  004537  021140          55$:      JSR     R5,RAND           ;GET RANDOM # FOR PATTERN
  4759  017270  013703  002126                    MOV     LONUM,R3          ;GET RANDOM PATTERN
  4760  017274  042703  177770                    BIC     #177770,R3        ;0,7
  4761  017300  006303                  56$:      ASL     R3                ;WORD OFFSET
  4762  017302  062703  024324                    ADD     #PATLST,R3        ;GET PATTERN LIST
  4763  017306  011303                            MOV     (R3),R3           ;GET LIST ADDRESS
  4764  017310  010337  002200                    MOV     R3,TEMP7          ;STOR FOR RECALL
  4765  017314  010321                            MOV     R3,(R1)+          ;LOAD IT
  4766  017316  005337  002176                    DEC     TEMP6             ;ACCOUNT FOR IT
```

```
017322  013703  002200          6$:     MOV     TEMP7,R3        ;PATTERN START
017326  012737  000020  002202          MOV     #16.,TEMP8      ;16 ENTRIES
017334  012321                  7$:     MOV     (R3)+,(R1)+     ;STORE PATTERN
017336  005337  002176                  DEC     TEMP6           ;DOWN COUNT
017342  001404                          BEQ     8$              ;DONE?
017344  005337  002202                  DEC     TEMP8           ;DONE WITH PATTERN
017350  001371                          BNE     7$              ;NO, GO BACK
017352  000763                          BR      6$              ;RESTART PATTERN
017354  162702  000200          8$:     SUB     #128.,R2        ;ANOTHER SECTOR TO USE
017360  003306                          BGT     2$              ;YES GO BACK
017362  012600                          MOV     (SP)+,R0        ;RESTORE REGISTERS
017364  012601                          MOV     (SP)+,R1
017366  012602                          MOV     (SP)+,R2
017370  012603                          MOV     (SP)+,R3
017372  000205                          RTS     R5

                                .SBTTL  RETRY LIMIT ROUTINE

                                ;RETRY BUMP, TWO RETURNS - CALL +2 - RETRY EXCEEDED
                                ;                          CALL +4 - CONTINUE RETRY

017374  026437  000036  007470  RCNT:   CMP     RETRY(R4),LIMIT ;LIMIT REACHED?
017402  001403                          BEQ     1$              ;YES TAKE FIRST RETURN
017404  005264  000036                  INC     RETRY(R4)       ;ACCOUNT FOR RETRY
017410  005725                          TST     (R5)+           ;NEXT RETURN
017412  000205                  1$:     RTS     R5              ;RETURN

                                .SBTTL  LIST OF FUNCTION ROUTINES

                                ;WE GO THRU THIS LIST WHEN CALLED IN "GETFNC"
                                ;LIST IS IN NUMERICAL ORDER 1-6 (CONTROLLER RESET - READ)

017414  000000                  LIST:   .WORD   0
017416  013662                          WRTFNC                  ;WRITE DATA
017420  013304                          GSTFNC                  ;GET STATUS
017422  013324                          SKFNC                   ;SEEK FUNCTION
017424  013324                          SKFNC                   ;SEEK FUNCTION
017426  013662                          WRTFNC                  ;WRITE DATA
017430  013732                          RDDFNC                  ;READ DATA

                                .SBTTL  BAD SECTOR FILE ROUTINE

                                ;ROUTINE TO RECOVER BAD SECTOR FILE AND SAVE IT FOR
                                ;COMPARISON UPON ERROR ON READS/WRITES.  WE WILL ONLY
                                ;RESERVE SPACE FOR 16 BAD SECTORS PER DRIVE.
                                ;WE WILL ISSUE A DRIVE RESET FIRST, READ HEADER, POSITION
                                ;TO LAST TRACK (CYLINDER 255, SURFACE 1) AND READ IN
                                ;THE FIRST SECTOR FOR FACTORY BAD, AND THE 20TH FOR
                                ;FIELD BAD SECTORS.  R4 WILL CONTAIN THE BUFFER POINTER
                                ;TO THE DRIVE WE WANT TO READ
                                ;
                                ;CALL   JSR     R5,RDBDSC


017432  010046                  RDBDSC: MOV     R0,-(SP)        ;SAVE PEGISTERS
017434  010146                          MOV     R1,-(SP)        ;
```

```
 017436  010246                              MOV     R2,-(SP)
 017440  010346                              MOV     R3,-(SP)
 017442  004537  021062          21$:        JSR     R5,ISDRST
 017446  012764  000010  000044              MOV     #RDHDR,FUNC (R4);READ HEADER TO FIND POSITION
 017454  004537  014042                      JSR     R5,LDFUNC        ;ON  DISK
 017460  004537  020776                      JSR     R5,WTRDY

 017464  016300  000006                      MOV     MP(R3),R0        ;GET HEADER AND CALCULATE
 017470  043700  002130                      BIC     CYLMSK,R0        ;DIFFERENCE TO GET TO
 017474  012701  077600                      MOV     #77600,R1        ;BAD SECTER FILE, AND GO
 017500  160001                              SUB     R0,R1            ;THERE
 017502  010164  000040                      MOV     R1,BDA(R4)
 017506  052764  000025  000040              BIS     #SKHS!SIGN!MK,BDA(R4)
 017514  012764  000006  000044              MOV     #SEEK,FUNC(R4)
 017522  004537  014042                      JSR     R5,LDFUNC
 017526  004537  020776                      JSR     R5,WTRDY
 017532  012764  000010  000044              MOV     #RDHDR,FUNC(R4)
 017540  004537  014042                      JSR     R5,LDFUNC
 017544  004537  020776                      JSR     R5,WTRDY
 017550  016300  000006                      MOV     MP(R3),R0
 017554  042700  000077                      BIC     #77,R0
 017560  022700  077700                      CMP     #77700,R0
 017564  001326                              BNE     21$

 017566  012764  077700  000040              MOV     #77700,BDA(R4)   ;SETUP AND READ IN THE
 017574  012764  177400  000042              MOV     #-256.,BMP(R4)   ;BAD SECTOR FILE ON SECTOR
 017602  012764  000014  000044              MOV     #READ,FUNC(R4)   ;0

 017610  005037  002170                      CLR     TEMP3            ;MANUFACTURING/FIELD FILE SWITCH
 017614  012737  003510  002116              MOV     #HWSEC,WHY       ;START WITH MANUFACTURING BAD
 017622  016402  000112                      MOV     BSECPT(R4),R2    ;INITIALIZE LIST TO ALL 1'S
 017626  012700  000020                      MOV     #16.,R0          ;SIXTEEN ENTRIES
 017632  012722  177777          11$:        MOV     #-1,(R2)+
 017636  005300                              DEC     R0
 017640  001374                              BNE     11$

 017642  016402  000112                      MOV     BSECPT(R4),R2    ;GET LIST TO STORE
 017646  012700  000020                      MOV     #16.,R0          ;SIXTEEN ENTRIES
 017652  004537  014042          4$:         JSR     R5,LDFUNC
 017656  004537  020776                      JSR     R5,WTRDY

 017662  005774  000104                      TST     @DCS(R4)         ;WAS THE READ GOOD?
 017666  100025                              BPL     3$               ;YES

 017670  004537  021062                      JSR     R5,ISDRST
 017674  062764  000004  000040              ADD     #4,BDA(R4)       ;NO, NEXT SECTOR
 017702  005737  002170                      TST     TEMP3            ;MANUFACTURING OR FIELD BAD
 017706  001410                              BEQ     5$               ;MANUFACTURING
 017710  012737  003530  002116              MOV     #SWSEC,WHY       ;FIELD BAD
 017716  022764  077750  000040              CMP     #77750,BDA(R4)   ;AT END OF FIELD BAD?
 017724  001352                              BNE     4$               ;NO, GO BACK FOR NEXT
 017726  000470                              BR      6$
 017730  026427  000040  077724  5$:         CMP     BDA(R4),#77724   ;AT END OF MANUFACTURING BAD
 017736  001345                              BNE     4$               ;AT END OF BAD FACTORY SECTION
 017740  000463                              BR      6$               ;YES, REPORT ERROR

```

```
017742  017401  000110          3$:     MOV     @BBA(R4),R1         ;START OF LIST
017746  012164  000100                  MOV     (R1)+,SERNM1(R4)    ;GET LOW PART OF SERIAL #
017752  012164  000102                  MOV     (R1)+,SERNM2(R4)    ;GET HIGH PART OF SERIAL #
017756  022121                          CMP     (R1)+,(R1)+         ;SKIP PAST JUNK
017760  012137  002164          1$:     MOV     (R1)+,TEMP1         ;GET CYLINDER
017764  100437                          BMI     2$                  ;IF MINUS END OF BAD SECTORS
017766  012137  002166                  MOV     (R1)+,TEMP2         ;GET TRACK AND CYLINDER
017772  000337  002164                  SWAB    TEMP1               ;PUT CYLINDER IN HIGH BYTE
017776  006237  002164                  ASR     TEMP1               ;ALIGN IT
020002  013712  002164                  MOV     TEMP1,(R2)          ;STORE OFF CYLINDER PART
020006  013737  002166  002164          MOV     TEMP2,TEMP1         ;GET SECTOR
020014  042737  177700  002164          BIC     #177700,TEMP1       ;LEAVE ONLY SECTOR
020022  053712  002164                  BIS     TEMP1,(R2)          ;SET IN SECTOR BITS
020026  042737  177377  002166          BIC     #177377,TEMP2
020034  006237  002166                  ASR     TEMP2
020040  006237  002166                  ASR     TEMP2
020044  053722  002166                  BIS     TEMP2,(R2)+         ;SET IN HEAD
020050  005300                          DEC     R0
020052  001342                          BNE     1$
020054  012737  003457  002116          MOV     #MBDMSC,WHY         ;MORE THAN 16 BAD SECTORS
020062  000412                          BR      6$

020064  005737  002170          2$:     TST     TEMP3               ;SWITCH TO FIELD BAD OR QUIT
020070  001011                          BNE     7$                  ;QUIT, 7$
020072  012764  077724  000040          MOV     #77724,BDA(R4)      ;SWITCH TO FIELD BAD
020100  012737  000001  002170          MOV     #1,TEMP3            ;SET TO QUIT NEXT TIME THRU
020106  000661                          BR      4$

020110  004537  020144          6$:     JSR     R5,DRDRV            ;DROP THE DRIVE
020114  004537  022374          7$:     JSR     R5,HDHOME           ;BRINGS HEADS HOME
020120  012603                  9$:     MOV     (SP)+,R3
020122  012602                          MOV     (SP)+,R2
020124  012601                          MOV     (SP)+,R1
020126  012600                          MOV     (SP)+,R0
020130  000205                          RTS     R5

020132  004537  020144          8$:     JSR     R5,DRDRV
020136  000770                          BR      9$


                                ;






                                .SBTTL  ROUTINE TO DROP DRIVE
                                ;
                                ;ROUTINE TO DROP A DRIVE FROM RUNNING
                                ;R4 HAS BUFFER POINTER OF DRIVE TO DROP
                                ;WE CLEAR BIT IN "DRUT", NOT "DRPRS"
                                ;

020140  005237  002310          ODRDRV: INC     OPCALL
020144  010146                  DRDRV:  MOV     R1,-(SP)
020146  010246                          MOV     R2,-(SP)            ;SAVE REGISTERS
020150  010346                          MOV     R3,-(SP)
```

```
020152  005237  002312                  INC     INCALL
020156  005003                          CLR     R3
020160  012702  024752                  MOV     #DRBUF,R2       ;START OF DRIVE BUFFERS
020164  012701  000001                  MOV     #1,R1           ;MASK
020170  020402                  1$:     CMP     R4,R2           ;IS THIS THE DRIVE?
020172  001405                          BEQ     2$              ;YES GO DROP IT
020174  005203                          INC     R3
020176  006301                          ASL     R1              ;NO SHIFT MASK
020200  062702  000122                  ADD     #PRPOS+2,R2     ;NEXT BUFFER
020204  000771                          BR      1$              ;GO BACK

020206  005737  002310          2$:     TST     OPCALL
020212  001002                          BNE     6$
020214                                  DODU    R3
020214  010300                          MOV     R3,R0
020216  104053                          EMT     C$DODU
020220  005037  002312          6$:     CLR     INCALL
020224  005037  002310                  CLR     OPCALL
020230  113764  002232  000070          MOVB    HOUR,DPHOUR(R4) ;TIME AT WHICH IT WAS DROPPED
020236  113764  002230  000071          MOVB    MINUTE,DPMIN(R4)  ;HOUR/MINUTE
020244  001002                          BNE     3$              ;IF MINUTE 0,
020246  105264  000071                  INCB    DPMIN(R4)       ;MAKE 1.
020252  140137  002120          3$:     BICB    R1,DRUT         ;CLEAR THE DRIVE FROM BIT MAP
020256                                  PRINTF  #FMT7,#TIME,HOUR,MINUTE,SECOND,#MRLCS,DCS(R4),#DRNM
020256  012746  003620                  MOV     #DRNM,-(SP)
020262  016446  000104                  MOV     DCS(R4),-(SP)
020266  012746  002323                  MOV     #MRLCS,-(SP)
020272  013746  002226                  MOV     SECOND,-(SP)
020276  013746  002230                  MOV     MINUTE,-(SP)
020302  013746  002232                  MOV     HOUR,-(SP)
020306  012746  002314                  MOV     #TIME,-(SP)
020312  012746  006072                  MOV     #FMT7,-(SP)
020316  012746  000010                  MOV     #10,-(SP)
020322  010600                          MOV     SP,R0
020324  104017                          EMT     C$PNTF
020326  062706  000022                  ADD     #22,SP
020332                                  PRINTF  #FMT7A,<B,DRSEL+1(R4)>,#DROP,WHY
020332  013746  002116                  MOV     WHY,-(SP)
020336  012746  004067                  MOV     #DROP,-(SP)
020342  005046                          CLR     -(SP)
020344  156416  000107                  BISB    DRSEL+1(R4),(SP)
020350  012746  006125                  MOV     #FMT7A,-(SP)
020354  012746  000004                  MOV     #4,-(SP)
020360  010600                          MOV     SP,R0
020362  104017                          EMT     C$PNTF
020364  062706  000012                  ADD     #12,SP
020370                                  PRINTF  #FMTS1
020370  012746  006703                  MOV     #FMTS1,-(SP)
020374  012746  000001                  MOV     #1,-(SP)
020400  010600                          MOV     SP,R0
020402  104017                          EMT     C$PNTF
020404  062706  000004                  ADD     #4,SP

020410  004737  011634                  JSR     PC,REPORT

020414  012603                          MOV     (SP)+,R3
```

```
     020416  012602                          MOV     (SP)+,R2        ;RESTORE REGISTERS
     020420  012601                          MOV     (SP)+,R1

     020422  000205                          RTS     R5

                                     .SBTTL  ROUTINE TO CHECK DATA

                                     ;ROUTINE TO CHECK DATA ON READ

     020424  005737  007504          CKDATA: TST     CMRD            ;DO WE WANT TO CHECK ANY?
     020430  001001                          BNE     97$             ;YES CONTINUE
     020432  000205                          RTS     R5              ;NO, EXIT
     020434                          97$:    SETPRI  #340
  (3)020434  012700  000340                  MOV     #340,R0
  (3)020440  104041                          EMT     C$SPRI
     020442  017402  000110                  MOV     @BBA(R4),R2     ;BUFFER START
     020446  016437  000042  002164          MOV     BMP(R4),TEMP1   ;WORDS READ IN
     020454  005437  002164                  NEG     TEMP1           ;MAKE POSITIVE
     020460  013737  007506  002166          MOV     DELMT,TEMP2     ;# ERRORS TO BE PRINTED
     020466  005037  002160                  CLR     DECNT           ;INIT ERROR COUNT
     020472  013737  007504  002170          MOV     CMRD,TEMP3      ;# WORDS TO BE COMPARED
     020500  012737  000176  002162  96$:    MOV     #126.,TEMP0     ;126 WORDS
     020506  012201                          MOV     (R2)+,R1        ;NON-ZERO WORDS
     020510  005337  002164                  DEC     TEMP1
     020514  001516                          BEQ     CEND
     020516  005301                          DEC     R1
     020520  012237  002172                  MOV     (R2)+,TEMP4     ;PATTERN ADDRESS

                                     ;MAKE SURE PATTERN ADDRESS IS LEGAL

     020524  012700  024324                  MOV     #PATLST,R0      ;GET LIST OF PATTERNS
     020530  012703  000010                  MOV     #8.,R3          ;ONLY EIGHT
     020534  022037  002172          98$:    CMP     (R0)+,TEMP4     ;FOUND IT YET
     020540  001412                          BEQ     99$             ;YES, CONTINUE
     020542  005303                          DEC     R3              ;NO, EXHAUST LIST YET
     020544  001373                          BNE     98$             ;NO, GO BACK

     020546  024242                          CMP     -(R2),-(R2)
     020550                                  ERRHRD  180.,NOREV,ERR13
  (3)020550  104463                          TRAP    T$ERCODE
  (5)020552  000264                          .WORD   180
  (5)020554  003420                          .WORD   NOREV
  (5)020556  005156                          .WORD   ERR13
     020560  004537  023514                  JSR     R5,STDMP
     020564  000205                          RTS     R5

     020566  005301                  99$:    DEC     R1              ;ACCOUNT FOR PATTERN ADDRESS
     020570  013703  002172                  MOV     TEMP4,R3        ;GET ADDRESS
     020574  005337  002164                  DEC     TEMP1           ;ACCOUNT ONCE AGAIN
     020600  012737  000020  002174          MOV     #16.,TEMP5      ;16 ENTRIES TO PATTERN
     020606  005737  002164          1$:     TST     TEMP1           ;ANY WORDS READIN LEFT?
     020612  001457                          BEQ     CEND            ;NO, GO TO END
     020614  005737  002170                  TST     TEMP3           ;HAVE WE EXHAUSTED COMPARE LIMIT?
     020620  001454                          BEQ     CEND            ;YES GO TO END
     020622  005701                          TST     R1              ;WE CHECKING PATTERN OR ZERO FILL?
     020624  001416                          BEQ     3$              ;ZERO FILL SKIP
```

```
 5013  020626  005301                            DEC     R1              ;PATTERN
 5014  020630  005737  002174                    TST     TEMP5           ;WITHIN PATTERN
 5015  020634  001005                            BNE     2$              ;YES SKIP
 5016  020636  013703  002172                    MOV     TEMP4,R3        ;NO, START OVER
 5017  020642  012737  000020  002174            MOV     #16.,TEMP5      ;16 ENTRIES
 5018  020650  012337  002216          2$:       MOV     (R3)+,GDDAT     ;GET PATTERN
 5019  020654  005337  002174                    DEC     TEMP5           ;DOWN COUNT
 5020  020660  000402                            BR      4$
 5021  020662  005037  002216          3$:       CLR     GDDAT           ;ZERO FILL
 5022  020666  023712  002216          4$:       CMP     GDDAT,(R2)      ;CORRECT DATA
 5023  020672  001415                            BEQ     5$              ;YES YES NEXT
 5024  020674  005237  002160                    INC     DECNT           ;DATA ERROR
 5025  020700  005264  000074                    INC     DATCER(R4)
 5026  020704  005737  002166                    TST     TEMP2           ;DO WE WANT TO PRINT IT
 5027  020710  001406                            BEQ     5$              ;NO,SKIP
 5028
 5029  020712                                    ERRHRD  180.,MDCER,ERR8
   (3) 020712  104463                            TRAP    T$ERCODE
   (5) 020714  000264                            .WORD   180
   (5) 020716  003024                            .WORD   MDCER
   (5) 020720  004650                            .WORD   ERR8
 5030  020722  005337  002166                    DEC     TEMP2           ;ACCOUNT FOR PRINT
 5031
 5032  020726  005337  002164          5$:       DEC     TEMP1           ;WORDS READ IN
 5033  020732  001407                            BEQ     CEND
 5034  020734  005722                            TST     (R2)+           ;NEXT WORD
 5035  020736  005337  002162                    DEC     TEMP0
 5036  020742  001656                            BEQ     96$
 5038  020744  005337  002170                DEC     TEMP3           ;WORDS TO CHECK
 5039  020750  000716                        BR      1$
 5040
 5041  020752  005737  002160        CEND:   TST     DECNT           ;DO WE WANT TO PRINT SUMMARY
 5042  020756  001406                        BEQ     1$              ;NO,EXIT
 5043
```

```
      020760  005464  000042                  NEG     BMP(R4)           ;MAKE POSITIVE WORD COUNT
      020764                                  ERRHRD  190.,MDCER,ERR6 ;DATA ERROR SUMMARY
  (3) 020764  104463                          TRAP    T$ERCODE
  (5) 020766  000276                          .WORD   190
  (5) 020770  003024                          .WORD   MDCER
  (5) 020772  004552                          .WORD   ERR6

      020774  000205                  1$:     RTS     R5

                                      .SBTTL  ROUTINE TO WAIT FOR CONTROLLER READY


                                      ;
                                      ;ROUTINE TO WAIT FOR CONTROLLER READY UNDER FLAG
                                      ;MODE.  USED IN INITIALIZE PORTION OF PROGRAM I.E.
                                      ;GETTING BAD SECTOR FILE, WRITING PACK INITIALLY

      020776  010046                  WTRDY:  MOV     R0,-(SP)          ;SAVE REGISTERS
      021000  010146                          MOV     R1,-(SP)
      021002  012701  001750                  MOV     #1000.,R1         ;WAIT A WHILE
      021006                          1$:     WAITUS  #2.
  (3) 021006  012700  000002                  MOV     #2.,R0
  (3) 021012  104027                          EMT     C$WTU
      021014  032774  000200  000104          BIT     #CRDY,@DCS(R4)  ;READY SET?
      021022  001006                          BNE     2$                ;YES, EXIT
      021024  005301                          DEC     R1                ;TIMED OUT?
      021026  001367                          BNE     1$                ;NO GO BACK

      021030                                  ERRDF   1002.,NOCRDY,ERR12
  (3) 021030  104462                          TRAP    T$ERCODE
  (5) 021032  001752                          .WORD   1002
  (5) 021034  002450                          .WORD   NOCRDY
  (5) 021036  005150                          .WORD   ERR12

      021040  012601                  2$:     MOV     (SP)+,R1          ;RESTORE REGISTERS
      021042  012600                          MOV     (SP)+,R0
      021044  000205                          RTS     R5


                                      .SBTTL  GET STATUS/DRIVE RESET ROUTINE

                                      ;ROUTINE TO ISSUE DRIVE RESET
                                      ;ALSO GET STATUS, R1 HAS STATUS IF GS
                                      ;USES R3, DOES NOT SAVE IT

      021046  016403  000104          GETDST: MOV     DCS(R4),R3
      021052  012763  000003  000004          MOV     #GSBIT,DA(R3)
      021060  000405                          BR      CSTUFF
      021062  016403  000104          ISDRST: MOV     DCS(R4),R3
      021066  012763  000013  000004          MOV     #DRST,DA(R3)
      021074  012763  000204  000000  CSTUFF: MOV     #CRDY!GSTAT,CS(R3)
      021102  056463  000106  000000          BIS     DRSEL(R4),CS(R3)
      021110  042763  000200  000000          BIC     #CRDY,CS(R3)
      021116  004537  020776                  JSR     R5,WTRDY
      021122  022763  000013  000004          CMP     #DRST,DA(R3)
      021130  001402                          BEQ     1$
      021132  016301  000006                  MOV     MP(R3),R1
```

```
 5091  021136  000205                  1$:     RTS     R5
 5092
 5093
 5094                                  .SBTTL  ROUTINE TO GENERATE A RANDOM NUMBER
 5095
 5096  021140  010146                  RAND:   MOV     R1,-(SP)
 5097  021142  010246                          MOV     R2,-(SP)
 5098  021144  010346                          MOV     R3,-(SP)
 5099
 5100  021146  013703  002126                  MOV     LONUM,R3
 5101  021152  013701  002124                  MOV     HINUM,R1
 5102  021156  012702  177771                  MOV     #-7,R2
 5103  021162  006303                  1$:     ASL     R3
 5104  021164  006101                          ROL     R1
 5105  021166  005202                          INC     R2
 5106  021170  001374                          BNE     1$
 5107  021172  063703  002126                  ADD     LONUM,R3
 5108  021176  005501                          ADC     R1
 5109  021200  063701  002124                  ADD     HINUM,R1
 5110  021204  062703  001057                  ADD     #1057,R3
 5111  021210  005501                          ADC     R1
 5112  021212  062701  047401                  ADD     #47401,R1
 5113  021216  010337  002124                  MOV     R3,HINUM
 5114  021222  010137  002126                  MOV     R1,LONUM
 5115  021226  012603                          MOV     (SP)+,R3
 5116  021230  012602                          MOV     (SP)+,R2
 5117  021232  012601                          MOV     (SP)+,R1
 5118  021234  000205                          RTS     R5
 5119
 5120                                  .SBTTL  ROUTINE TO WRITE PACKS INITIALLY
 5121
 5122                                  ;ROUTINE TO WRITE PACK WITH PATTERN, ALL TRACKS WILL BE
 5123                                  ;WRITTEN (EXCEPT BAD SECTOR TRACK)
 5124                                  ;FORMAT IS # OF WORDS (WORD 1), PATTERN ADDRESS (WORD 2)
 5125                                  ;PATTERN (WORDS 3 - 128)
 5126                                  ;WE WILL ATTEMPT TO WRITE MULTIPLE SECTORS AT A TIME
 5127                                  ;(MINIMUM 10 SECTORS) IF AN ERROR OCCURS WE WILL THEN
 5128                                  ;WRITE INDIVIDUAL SECTORS FOR THAT TRACK.  WE DO WRITES,
 5129                                  ;READS AND INCORE COMPARISONS TO VERIFY.
 5130
 5131                                  ;
 5132                                  ;CALL JSR R5,WRPACK
 5133
 5134  021236  010046                  WRPACK: MOV     R0,-(SP)                ;SAVE REGISTERS
 5135  021240  010146                          MOV     R1,-(SP)
 5136  021242  010246                          MOV     R2,-(SP)
 5137  021244  010346                          MOV     R3,-(SP)
 5138  021246  016446  000110                  MOV     BBA(R4),-(SP)
 5139  021252                                  PRINTF  #FMT18,#MSWRPK
  (8)  021252  012746  004171                  MOV     #MSWRPK,-(SP)
  (7)  021256  012746  006676                  MOV     #FMT18,-(SP)
  (6)  021262  012746  000002                  MOV     #2,-(SP)
  (3)  021266  010600                          MOV     SP,R0
  (4)  021270  104017                          EMT     C$PNTF
  (4)  021272  062706  000006                  ADD     #6,SP
 5140  021276                                  PRINTF  #FMT17,#MRLCS,DCS(R4),#DRNM,<B.DRSEL+1(R4)>
 (11) 021276  005046                               CLR     -(SP)
 (11) 021300  156416  000107                       BISB    DRSEL+1(R4),(SP)
 (10) 021304  012746  003620                       MOV     #DRNM,-(SP)
 (9)  021310  016446  000104                       MOV     DCS(R4),-(SP)
 (8)  021314  012746  002323                       MOV     #MRLCS,-(SP)
 (7)  021320  012746  005647                       MOV     #FMT17,-(SP)
 (6)  021324  012746  000005                       MOV     #5,-(SP)
 (3)  021330  010600                               MOV     SP,R0
 (4)  021332  104017                               EMT     C$PNTF
 (4)  021334  062706  000014                       ADD     #14,SP
      021340  004537  022374                       JSR     R5,HDHOME         ;HEADS HOME


                                           ;
                                           ;NOW ACTUALLY WRITE DATA OUT ON PACK, WILL NOT WRITE LAST
                                           ;TRACK
                                           ;

      021344  005037  002164                       CLR     TEMP1             ;TEMP1=HEAD
      021350  005001                               CLR     R1                ;R1=CYL
      021352  022701  077600               CONWR:  CMP     #77600,R1         ;CYL=255?
      021356  001014                               BNE     STWRT             ;NO GO WRITE TRACK
      021360  005737  002164                       TST     TEMP1             ;YES, CHECK IF HEAD = 1?
      021364  001411                               BEQ     STWRT             ;HEAD = 0 GO WRITE
      021366  004537  022374               ENDWR:  JSR     R5,HDHOME         ;HEADS HOME
      021372  012664  000110                       MOV     (SP)+,BBA(R4)
      021376  012603                               MOV     (SP)+,R3
      021400  012602                               MOV     (SP)+,R2
      021402  012601                               MOV     (SP)+,R1
      021404  012600                               MOV     (SP)+,R0
      021406  000205                               RTS     R5                ;END EXIT

                                           ;THIS PORTION WILL WRITE THE PACK USING MULTIPLE SECTORS IF A
                                           ;ERROR OCCURS WE WILL GO TO 2$ AND INDIVIDUAL SECTORS.

      021410  005002                       STWRT:  CLR     R2                ;INITIAL SECTOR 0
      021412  012764  002252  000110               MOV     #BUF1,BBA(R4)     ;BUFFER START
      021420  012764  175400  000042               MOV     #-1280.,BMF(R4)   ;10 SECTORS
      021426  004537  017154                       JSR     R5,WRBUF          ;WRITE BUFFER INTO MEMORY
      021432  010164  000040               201$:   MOV     R1,BDA(R4)        ;SET UP SECTOR
      021436  053764  002164  000040               BIS     TEMP1,BDA(R4)
      021444  050264  000040                       BIS     R2,BDA(R4)
      021450  012764  002252  000110               MOV     #BUF1,BBA(R4)     ;SET UP TO WRITE
      021456  012764  000012  000044               MOV     #WRITE,FUNC(R4)   ;WRITE
      021464  004537  014042                       JSR     R5,LDFUNC
      021470  004537  020776                       JSR     R5,WTRDY          ;WAIT FOR READY
      021474  005774  000104                       TST     @DCS(R4)          ;ERROR
      021500  100003                               BPL     203$
      021502  004537  021062               205$:   JSR     R5,ISDRST
      021506  000421                               BR      2$

      021510  012764  000002  000044       203$    MOV     #WRCHK,FUNC(R4)
      021516  004537  014042                       JSR     R5,LDFUNC
      021522  004537  020776                       JSR     R5,WTRDY
      021526  005774  000104                       TST     @DCS(R4)          ;ERROR
      021532  100763                               BMI     205$              ;YES GO DO SECTORS INDIVIDUALLY

```

```

021534  062702  000012                          ADD     #10.,R2          ;NEXT GROUP
021540  022702  000050                          CMP     #40.,R2          ;DONE?
021544  001332                                  BNE     201$             ;NO, GO BACK
021546  000137  022050                          JMP     952$             ;YES NEXT TRACK

                                ;IF AN ERROR OCCURS THEN WE COME HERE AND DO THE TRACK SECTOR
                                ;BY SECTOR.

021552  005002                          2$:     CLR     R2               ;R2 = SECTOR

021554  012764  177600  000042                  MOV     #-128.,BMP(R4)   ;LOAD WORD COUNT
021562  010164  000040                  3$:     MOV     R1,BDA(R4)       ;SETUP DISK ADDRESS
021566  053764  002164  000040                  BIS     TEMP1,BDA(R4)
021574  050264  000040                          BIS     R2,BDA(R4)

021600  012764  002252  000110                  MOV     #BUF1,BBA(R4)
021606  004537  017154                          JSR     R5,WRBUF         ;WRITE A BUFFER
021612  005037  002114                  91$:    CLR     RWCNT            ;CLEAR RETRYS OUT
021616  005037  002160                  98$:    CLR     DECNT            ;
021622  012764  000012  000044          96$:    MOV     #WRITE,FUNC(R4)  ;WRITE FUNCTION
021630  004537  014042                          JSR     R5,LDFUNC
021634  004537  020776                          JSR     R5,WTRDY         ;WAIT FOR WRITE TO FINISH

021640  005774  000104                          TST     @DCS(R4)         ;ERROR ON WRITE?
021644  100023                                  BPL     85$              ;NO, GO READ

021646  004537  021062                          JSR     R5,ISDRST
021652  016437  000040  002156                  MOV     BDA(R4),CHKSEC   ;YES, CHECK IF SECTOR IS IN
021660  004537  023640                          JSR     R5,CKBDSC        ;BAD SECTOR FILE
021664  005737  002154                          TST     HDRFND           ;IF SET, IT WAS
021670  001050                                  BNE     95$              ;YES GO TO NEXT SECTOR

021672  005237  002160                          INC     DECNT            ;NO, GIVE IT ONE MORE TRY
021676  023727  002160  000002                  CMP     DECNT,#2.        ;IT MAY HAVE BEEN NOISE.
021704  001346                                  BNE     96$


021706  004537  022134                          JSR     R5,INBAD
021712  000437                                  BR      95$


021714  005037  002112                  85$:    CLR     RECNT   ;        ;CLEAR RETRY COUNT
021720  012764  000002  000044          80$:    MOV     #WRCHK,FUNC(R4)  ;
021726  004537  014042                          JSR     R5,LDFUNC
021732  004537  020776                          JSR     R5,WTRDY

021736  005774  000104                          TST     @DCS(R4)         ;ERROR ON READ
021742  100023                                  BPL     81$              ;NO, GO COMPARE
021744  004537  021062                          JSR     R5,ISDRST

021750  016437  000040  002156                  MOV     BDA(R4),CHKSEC   ;CHECK IF SECTOR IS
021756  004537  023640                          JSR     R5,CKBDSC        ;A KNOWN BAD SECTOR
021762  005737  002154                          TST     HDRFND           ;IT WAS THEN
021766  001011                                  BNE     95$              ;GO TO NEXT SECTOR

```

```
 5243  021770  005237  002112                      INC     RECNT                   ;GIVE IT ANOTHER CHANCE
 5244  021774  023727  002112  000002              CMP     RECNT,#2.
 5245  022002  001346                              BNE     80$
 5246
 5247  022004  004537  022134                      JSR     R5,INBAD
 5248  022010  000400                              BR      95$
 5249
 5250  022012                              81$:
 5251
 5252  022012  062702  000012              95$:    ADD     #10.,R2                 ;NEXT SECTOR (OFFSET BY 10)
 5253  022016  020227  000047                      CMP     R2,#39.                 ;DONE WITH TRACK?
 5254  022022  003002                              BGT     951$                    ;YES NEXT TRACK
 5255  022024  000137  021562                      JMP     3$                      ;NO GO BACK FOR NEXT SECTOR
 5256  022030                              951$:
 5257  022030  005202                              INC     R2                      ;NEXT SECTOR
 5258  022032  162702  000050                      SUB     #40.,R2                 ;DONE WITH TRACK?
 5259  022036  020227  000012                      CMP     R2,#10.                 ;
 5260  022042  001402                              BEQ     952$                    ;YES
 5261  022044  000137  021562                      JMP     3$                      ;NO
 5262  022050                              952$:
 5263
 5264  022050  005737  002164                      TST     TEMP1                   ;WHICH SURFACE?
 5265  022054  001420                              BEQ     5$                      ;TOP (0), BRANCH
 5266
 5267  022056  005037  002164                      CLR     TEMP1                   ;BOTTOM, SWITCH TO TOP WITH
 5268  022062  062701  000200                      ADD     #200,R1
 5269  022066  012764  000205  000040              MOV     #205,BDA(R4)            ;SEEK, GO IN ALSO
 5270  022074  012764  000006  000044      4$:     MOV     #SEEK,FUNC(R4)          ;GO SEEK
 5271  022102  004537  014042                      JSR     R5,LDFUNC
 5272  022106  004537  020776                      JSR     R5,WTRDY
 5273
 5274  022112  000137  021352                      JMP     CONWR
 5275
 5276  022116  012737  000100  002164      5$:     MOV     #HEAD,TEMP1             ;WAS TOP, MAKE BOTTOM
 5277  022124  012764  000021  000040              MOV     #21,BDA(R4)
 5278  022132  000760                              BR      4$
 5279
 5280
 5281  022134  016337  000000  002234      INBAD:  MOV     CS(R3),E.CS
 5282  022142  016337  000002  002236              MOV     BA(R3),E.BA
 5283  022150  016337  000004  002240              MOV     DA(R3),E.DA
 5284  022156  016337  000006  002242              MOV     MP(R3),E.MP
 5285  022164  016337  000006  002244              MOV     MP(R3),E.MP1
 5286  022172  016337  000006  002246              MOV     MP(R3),E.MP2
 5287  022200                                      ERRHRD  199.,NWRTS,ERR13
  (3)  022200  104463                              TRAP    T$ERCODE
  (5)  022202  000307                              .WORD   199
  (5)  022204  002525                              .WORD   NWRTS
  (5)  022206  005156                              .WORD   ERR13
 5288  022210  005264  000012                      INC     ERRCNT(R4)
 5289  022214  005737  007532                      TST     T.DRP                   ;ARE WE CCUNTING ERRORS
 5290  022220  001413                              BEQ     2$                      ;NO
 5291  022222  026437  000012  007472              CMP     ERRCNT(R4),ERLMT;PAST IT
 5292  022230  103407                              BLO     2$                      ;NO
 5293  022232  012737  003111  002116              MOV     #ERLMTM,WHY
 5294  022240  004537  020144                      JSR     R5,DRDRV
```

```
     022244  012705  021366                          MOV     #ENDWR,R5

     022250  000205                          2$:     RTS     R5
                                             .SBTTL  ROUTINE FOR SYSTEM CLOCK

                                             ;ROUTINE TO READ SYSTEM CLOCK
                                             ;USES 'REGTIM' FROM DIAGNOSTIC SUPERVISOR

     022252  005737  002250                  GETSYS: TST     SYSCLK                  ;DO WE HAVE A CLOCK
     022256  001002                                  BNE     4$                      ;YES, GO SERVICE IT
     022260                                          BREAK                           ;NO, CALL SUPER FOR  C
 (3) 022260  104022                                  EMT     C$BRK
     022262  000205                                  RTS     R5                      ;EXIT
     022264                                  4$:     REQTIM  R0                      ;GET PRESENT TIME
 (3) 022264  104045                                  EMT     C$REQTIM
     022266  020037  002224                  1$:     CMP     R0,LSTTIM               ;HAS IT MOVED
     022272  001437                                  BEQ     3$                      ;NO MOVEMEMT SINCE LAST CALL
     022274  013701  002224                          MOV     LSTTIM,R1               ;CALCULATE DIFFERENCE
     022300  010037  002224                          MOV     R0,LSTTIM               ;AND FIX ACCORDINGLY
     022304  160100                                  SUB     R1,R0
     022306  060037  002226                  2$:     ADD     R0,SECOND               ;BUMP SECONDS
     022312  022737  000074  002226                  CMP     #60.,SECOND             ;SECONDS OVERFLOW
     022320  003024                                  BGT     3$
     022322  162737  000074  002226          7$:     SUB     #60.,SECOND
     022330  005237  002222                          INC     INTERVAL                ;TIME BETWEEN REPORTS
     022334  005237  002230                          INC     MINUTE                  ;BUMP MINUTES
     022340  022737  000074  002226                  CMP     #60.,SECOND
     022346  002765                                  BLT     7$
     022350  022737  000074  002230                  CMP     #60.,MINUTE
     022356  003005                                  BGT     3$
     022360  005237  002232                          INC     HOUR
     022364  162737  000074  002230                  SUB     #60.,MINUTE
     022372  000205                          3$:     RTS     R5

                                             .SBTTL  HEADS HOME ROUTINE

                                             ;ROUTINE TO BRING HEADS OVER TRACK 0

     022374  010046                          HDHOME: MOV     R0,-(SP)                ;SAVE R0
     022376  012764  000010  000044                  MOV     #RDHDR,FUNC(R4) ;READ HEADER
     022404  004537  014042                          JSR     R5,LDFUNC               ;GO DO IT.
     022410  004537  020776                          JSR     R5,WTRDY

     022414  016300  000006                          MOV     MP(R3),R0               ;GET HEADER
     022420  042700  000177                          BIC     #177,R0                 ;ONLY CYLINDER
     022424  010064  000040                          MOV     R0,BDA(R4)              ;MOVE IT TO BUFFERED DA
     022430  052764  000001  000040                  BIS     #MK,BDA(R4)             ;SET MARKER
     022436  012764  000006  000044                  MOV     #SEEK,FUNC(R4)          ;LOAD SEEK
     022444  004537  014042                          JSR     R5,LDFUNC               ;SEEK!
     022450  004537  020776                          JSR     R5,WTRDY                ;WAIT.
     022454  016464  000120  000050                  MOV     PRPOS(R4),LSTHDR(R4)
     022462  005064  000120                          CLR     PRPOS(R4)               ;SET BUFFER TO HOME
     022466  012600                                  MOV     (SP)+,R0
     022470  000205                                  RTS     R5

                                             .SBTTL  RANDOM WC AND DA ROUTINE
```

```

                                           ;ROUTINE TO GET RANDOM SECTOR AND WORD COUNT FOR R/W TRANSFER.
                                           ;SECTOR IS CHOSEN BETWEEN MIN/MAX LIMITS. WORD COUNT IS BETWEEN
                                           ;MIN/MAX WORD COUNT. WORD COUNT WILL BE ADJUSTED NOT TO CAUSE
                                           ;TRACK OVERFLOW IF HIGH SECTORS ARE CHOSEN....
                                           ;R4 HAS BUFFER OF DRIVE WE'RE WORKING WITH
                                           ;ON EXIT - BMP(R4) HAS WORD COUNT.
                                           ;         - BDA(R4)      HAS     DISK ADDRESS
                                           ;
      022472  023737  007524  007526  GWCDA:  CMP     T.MXS,T.MNS       ;MIN MAX SECTORS EQUAL
      022500  001003                          BNE     99$               ;NO, CALCULATE ONE
      022502  013702  007524                  MOV     T.MXS,R2          ;LOAD SECTOR
      022506  000421                          BR      5$                ;GO GET WC
      022510  004537  021140          99$:    JSR     R5,RAND           ;GET RANDOM # FOR SECTOR
      022514  013702  002126                  MOV     LONUM,R2
      022520  042702  177700          1$:     BIC     #177700,R2        ;0-77 ONLY
      022524  023702  007524                  CMP     T.MXS,R2          ;R2 LOWER THAN MAX
      022530  103003                          BHIS    3$                ;BRANCH IF YES
      022532  006202                          ASR     R2                ;HALF IT
      022534  005202                          INC     R2                ;INC SO NOT 0
      022536  000770                          BR      1$
      022540  020237  007526          3$:     CMP     R2,T.MNS          ;MIN OKAY
      022544  103002                          BHIS    5$
      022546  006102                          ROL     R2
      022550  000763                          BR      1$


                                           ;NOW GET WORD COUNT

      022552  005737  007560          5$:     TST     T.STIP
      022556  001003                          BNE     95$
      022560  013737  002256  007512          MOV     MAXWC,T.MXB
      022566  023737  002256  007512  95$:    CMP     MAXWC,T.MXB
      022574  103021                          BHIS    97$

      022576                                  PRINTF  #FMT13D,#OVER,T.MXB,MAXWC
 (10) 022576  013746  002256                  MOV     MAXWC,-(SP)
  (9) 022602  013746  007512                  MOV     T.MXB,-(SP)
  (8) 022606  012746  003242                  MOV     #OVER,-(SP)
  (7) 022612  012746  006446                  MOV     #FMT13D,-(SP)
  (6) 022616  012746  000004                  MOV     #4,-(SP)
  (3) 022622  010600                          MOV     SP,R0
  (4) 022624  104017                          EMT     C$PNTF
  (4) 022626  062706  000012                  ADD     #12,SP
      022632  013737  002256  007512          MOV     MAXWC,T.MXB

      022640  023737  007512  007534  97$:    CMP     T.MXB,T.MNB       ;MIN MAX EQUAL
      022646  003006                          BGT     6$
      022650  013737  007512  007534          MOV     T.MXB,T.MNB

      022656  013703  007512                  MOV     T.MXB,R3          ;YES SET WC
      022662  000421                          BR      9$
      022664  004537  021140          6$:     JSR     R5,RAND           ;GET RANDOM WORD COUNT
      022670  013703  002126                  MOV     LONUM,R3
      022674  042703  160000          7$.     BIC     #160000,R3        ;MAX!!!!!
      022700  023703  007512                  CMP     T.MXB,R3
```

```
022704  103003                            BHIS    8$
022706  006203                            ASR     R3
022710  005203                            INC     R3
022712  000770                            BR      7$
022714  020337  007534          8$:       CMP     R3,T.MNB
022720  103002                            BHIS    9$
022722  006103                            ROL     R3
022724  000763                            BR      7$

                                ;NOW WE HAVE SECTOR AND WORD COUNT, CHECK THAT WORD COUNT WILL FIT ON SECTOR
                                ;IF NOT LOWER SECTOR START


022726  012701  000050          9$:       MOV     #40.,R1             ;SETUP FOR FOURTY SECTORS
022732  005403                            NEG     R3                  ;MAKE WORD COUNT NEGATIVE
022734  010364  000042                    MOV     R3,BMP(R4)          ;LOAD WORD COUNT
022740  005301                  11$:      DEC     R1                  ;DOWN COUNT MINIMUM START SECT NEEDED
022742  062703  000200                    ADD     #128.,R3            ;ONE SECTOR'S WORTH
022746  100774                            BMI     11$                 ;STILL NEED ANOTHER SECTOR
022750  020201                            CMP     R2,R1               ;DID RANDOM SECTOR SUFFICE
022752  101401                            BLOS    12$                 ;BRANCH IF SUFFICED
022754  010102                            MOV     R1,R2               ;NO, THEN MAKE IT FIT
022756  016464  000120  000040  12$:      MOV     PRPOS(R4),BDA(R4)
022764  042764  000077  000040            BIC     #77,BDA(R4)
022772  050264  000040                    BIS     R2,BDA(R4)
022776  000205                            RTS     R5
















                                .SBTTL  ROUTINE TO DUMP BUFFER ON DCK

                                ;ROUTINE TO DUMP BUFFER ON DCK ERROR, TWO DUMPS ARE POSSIBLE
                                ;ONE WHERE WE CAN COMPARE WHAT IT SHOULD BE AND THE OTHER
                                ;WHEN WE CAN'T

023000  004737  005214          DMPBUF:   JSR     PC,LINE1


                                ;
                                ;CALCULATE THE STARTING BUS ADDRESS FOR THE COMPARE
                                ;

023004  012737  000200  002300            MOV     #128.,DWCNT1
023012  016400  000040                    MOV     BDA(R4),R0                ;GET STARTING BUS ADDRESS

```

```
 5453  023016  013701  002240                MOV     E.DA,R1                          ;GET PRESENT DISK ADDRESS
 5454  023022  042700  177700                BIC     #177700,R0                       ;SAVE SECTOR BITS
 5455  023026  042701  177700                BIC     #177700,R1                       ;
 5456  023032  010002                        MOV     R0,R2                            ;SAVE A COPY
 5457  023034  010103                        MOV     R1,R3                            ;SAVE ANOTHER
 5458  023036  160203                        SUB     R2,R3                            ;GET DIFF OF SECTORS
 5459  023040  005002                        CLR     R2                               ;CALCULATE WORD COUNT
 5460  023042  062702  000200        93$:    ADD     #128.,R2                         ;ONE SECTORS WORTH
 5461  023046  005303                        DEC     R3                               ;DONE
 5462  023050  001374                        BNE     93$                              ;NO
 5463  023052  016403  000042                MOV     BMP(R4),R3                       ;GET WORD COUNT
 5464  023056  005403                        NEG     R3                               ;MAKE IT POSITIVE
 5465  023060  020203                        CMP     R2,R3                            ;WORKING WITH FULL SECTOR
 5466  023062  003005                        BGT     94$                              ;NO, GO CALC PARTIAL SECTOR
 5467  023064  013702  002236                MOV     E.BA,R2                          ;PRESENT BUS ADDRESS
 5468  023070  162702  000400                SUB     #400,R2                          ;START OF COMPARE
 5469  023074  000412                        BR      96$                              ;GO COMPARE BUFFER
 5470  023076  160302                94$:    SUB     R3,R2                            ;GET SECTOR DIFF
 5471  023100  012700  000200                MOV     #128.,R0
 5472  023104  160200                        SUB     R2,R0
 5473  023106  010037  002300                MOV     R0,DWCNT1
 5474  023112  006300                        ASL     R0
 5475  023114  013702  002236                MOV     E.BA,R2
 5476  023120  160002                        SUB     R0.R2
 5477  023122                        96$:    PRINTB  #FMT13,#BUSAD,R2,#CRLDA,CHKSEC
  (11) 023122  013746  002156                MOV     CHKSEC,-(SP)
  (10) 023126  012746  002373                MOV     #CRLDA,-(SP)
   (9) 023132  010246                        MOV     R2,-(SP)
   (8) 023134  012746  003723                MOV     #BUSAD,-(SP)
   (7) 023140  012746  006431                MOV     #FMT13,-(SP)
   (6) 023144  012746  000005                MOV     #5,-(SP)
   (3) 023150  010600                        MOV     SP,R0
   (4) 023152  104014                        EMT     C$PNTB
   (4) 023154  062706  000014                ADD     #14,SP
 5478  023160  012700  024324                MOV     #PATLST,R0                       ;CHECK PATTERN LIST
 5479  023164  012701  000010                MOV     #8.,R1
 5480  023170  022062  000002        1$:     CMP     (R0)+,2(R2)
 5481  023174  001415                        BEQ     2$
 5482  023176  005301                        DEC     R1
 5483  023200  001373                        BNE     1$
 5484
 5485  023202                        3$:     PRINTB  #FMT14,#NOREV
   (8) 023202  012746  003420                MOV     #NOREV,-(SP)
   (7) 023206  012746  006473                MOV     #FMT14,-(SP)
   (6) 023212  012746  000002                MOV     #2,-(SP)
   (3) 023216  010600                        MOV     SP,R0
   (4) 023220  104014                        EMT     C$PNTB
   (4) 023222  062706  000006                ADD     #6,SP
 5486  023226  000532                        BR      STDMP
 5487
 5488  023230  021227  000200        2$.     CMP     (R2),#128.
 5489  023234  101362                        BHI     3$
 5490  023236  005037  002160                CLR     DECNT
 5491  023242  013701  007554                MOV     T.CLT,R1
 5492
 5493  023246  012237  002162                MOV     (R2)+,TEMPO      ;NONZERO WORD COUNT
```

```
  5494  023252  013737  002162  002276          MOV     TEMP0,DWCNT
  5495  023260  005437  002276                  NEG     DWCNT
  5496  023264  012237  002164                  MOV     (R2)+,TEMP1
  5497  023270  162737  000002  002162          SUB     #2,TEMP0
  5498  023276  012737  000002  002166          MOV     #2,TEMP2        ;WORD
  5499  023304  013703  002164                  MOV     TEMP1,R3        ;PATTERN ADDRESS
  5500  023310  012737  000020  002174          MOV     #16.,TEMP5      ;16 ENTRIES
  5501  023316  005737  002162          4$:     TST     TEMP0           ;ZERO OR PATTERN
  5502  023322  001417                          BEQ     6$              ;ZERO BRANCH
  5503  023324  005337  002162                  DEC     TEMP0
  5504  023330  005737  002174                  TST     TEMP5            ;WITHIN LIST
  5505  023334  001005                          BNE     5$
  5506  023336  012737  000020  002174          MOV     #16.,TEMP5
  5507  023344  013703  002164                  MOV     TEMP1,R3
  5508  023350  012337  002216          5$:     MOV     (R3)+,GDDAT
  5509  023354  005337  002174                  DEC     TEMP5
  5510  023360  000402                          BR      7$
  5511  023362  005037  002216          6$:     CLR     GDDAT
  5512  023366  005237  002276          7$:     INC     DWCNT
  5513  023372  021237  002216                  CMP     (R2),GDDAT
  5514  023376  001422                          BEQ     8$
  5515
  5516  023400  005237  002160                  INC     DECNT
  5517  023404  005701                          TST     R1
  5518  023406  001416                          BEQ     8$
  5519  023410  005301                          DEC     R1
  5520  023412                                  PRINTB  #FMT14B,TEMP2,GDDAT,(R2)
   (10) 023412  011246                          MOV     (R2),-(SP)
   (9)  023414  013746  002216                  MOV     GDDAT,-(SP)
   (8)  023420  013746  002166                  MOV     TEMP2,-(SP)
   (7)  023424  012746  006514                  MOV     #FMT14B,-(SP)
   (6)  023430  012746  000004                  MOV     #4,-(SP)
   (3)  023434  010600                          MOV     SP,R0
   (4)  023436  104014                          EMT     C$PNTB
   (4)  023440  062706  000012                  ADD     #12,SP
  5521
  5522  023444  005237  002166          8$:     INC     TEMP2
  5523  023450  005722                          TST     (R2)+
  5524  023452  023737  002300  002166          CMP     DWCNT1,TEMP2
  5525  023460  003716                          BLE     4$
  5526  023462                                  PRINTB  #FMT9A,DECNT,TEMP2
   (9)  023462  013746  002166                  MOV     TEMP2,-(SP)
   (8)  023466  013746  002160                  MOV     DECNT,-(SP)
   (7)  023472  012746  006214                  MOV     #FMT9A,-(SP)
   (6)  023476  012746  000003                  MOV     #3,-(SP)
   (3)  023502  010600                          MOV     SP,R0
   (4)  023504  104014                          EMT     C$PNTB
   (4)  023506  062706  000010                  ADD     #10,SP
  5527
  5528  023512  000205                          RTS     R5
  5529
  5530  023514  013701  007554          STDMP:  MOV     T.CLT,R1
  5531  023520  012703  000012                  MOV     #10.,R3
  5532  023524                          1$:     PRINTB  #FMT14A,(R2)
   (8)  023524  011246                          MOV     (R2),-(SP)
   (7)  023526  012746  006502                  MOV     #FMT14A,-(SP)
  (6)  023532  012746  000002                  MOV     #2,-(SP)
  (3)  023536  010600                          MOV     SP,R0
  (4)  023540  104014                          EMT     C$PNTB
  (4)  023542  062706  000006                  ADD     #6,SP
       023546  005722                          TST     (R2)+
       023550  005303                          DEC     R3
       023552  001012                          BNE     2$
       023554                                  PRINTB  #FMT14C
  (7)  023554  012746  006511                  MOV     #FMT14C,-(SP)
  (6)  023560  012746  000001                  MOV     #1,-(SP)
  (3)  023564  010600                          MOV     SP,R0
  (4)  023566  104014                          EMT     C$PNTB
  (4)  023570  062706  000004                  ADD     #4,SP
       023574  012703  000012                  MOV     #10.,R3
       023600  005337  002300          2$:     DEC     DWCNT1
       023604  001001                          BNE     3$
       023606  000205                          RTS     R5
       023610  005301                  3$:     DEC     R1
       023612  001344                          BNE     1$
       023614  000205                          RTS     R5



                                       ;
                                       ;ROUTINE TO CLEAR ALL DRIVE INFO, USED ON START OR
                                       ;RESTART IF CALLED.  CAN BE USED TO CLEAR INDIVIDUAL DRIVE
                                       ;INFO BY BITMAP FOLLOWING CALL
                                       ;CALL    JSR     R5,CLEAR
                                       ;


       023616  010446                  CLEAR:  MOV     R4,-(SP)                        ;SAVE R4
       023620  012704  024752                  MOV     #DRBUF,R4                       ;GET BUFFER STARTS
       023624  005024                  2$:     CLR     (R4)+                           ;CLEAR
       023626  020427  026172                  CMP     R4,#ENDBUF                      ;AT END OF BUFFERS
       023632  001374                          BNE     2$                              ;NO, GO TO 2$
       023634  012604                  4$:     MOV     (SP)+,R4                        ;RESTORE CURRENT BUFFER POINTER
       023636  000205                          RTS     R5                              ;EXIT

                                       .SBTTL  ROUTINE TO CHECK FOR BAD SECTOR

                                       ;ROUTINE TO MATCH BAD SECTOR.....BDA(R4) IS SECTOR WE ARE LOOKING
                                       ;FOR IN LIST POINTED TO BY BSECPT(R4)......HDRFND IS SET IF WE FIND IT.
                                       ;


       023640  005037  002154          CKBDSC: CLR     HDRFND          ;CLEAR FLAG
       023644  010046                          MOV     R0,-(SP)        ;SAVE R0
       023646  010146                          MOV     R1,-(SP)        ;SAVE R1
       023650  010246                          MOV     R2,-(SP)        ;SAVE   R2
       023652  010346                          MOV     R3,-(SP)        ;SAVE   R3
       023654  012700  000020                  MOV     #16.,R0         ;16 ENTRIES
       023660  016402  000112          1$:     MOV     BSECPT(R4),R2   ;GET WHERE WE'RE LOOKING
       023664  005712                  2$:     TST     (R2)            ;END
       023666  100411                          BMI     4$
       023670  023712  002156                  CMP     CHKSEC,(R2)     ;HAVE WE GOT A MATCH
  5580  023674  001404                              BEQ     3$                      ;THEN GO SET INDICATOR, ELSE
  5581  023676  005722                              TST     (R2)+
  5582  023700  005300                              DEC     R0
  5583  023702  001370                              BNE     2$
  5584  023704  000402                              BR      4$
  5585
  5586  023706  005237  002154              3$:     INC     HDRFND                  ;SET FLAG FOUND
  5587
  5588  023712  012603                      4$:     MOV     (SP)+,R3
  5589  023714  012602                              MOV     (SP)+,R2
  5590  023716  012601                              MOV     (SP)+,R1
  5591  023720  012600                              MOV     (SP)+,R0
  5592  023722  000205                              RTS     R5
  5593
  5594
  5595                                      ;BUFFER TO STORE BAD SECTOR LISTS
  5596
  5597  023724  000020                      BSEC0:  .BLKW   16.
  5598  023764  000020                      BSEC1:  .BLKW   16.
  5599  024024  000020                      BSEC2:  .BLKW   16.
  5600  024064  000020                      BSEC3:  .BLKW   16.
  5601  024124  000020                      BSEC4:  .BLKW   16.
  5602  024164  000020                      BSEC5:  .BLKW   16.
  5603  024224  000020                      BSEC6:  .BLKW   16
  5604  024264  000020                      BSEC7:  .BLKW   16.
  5605
  5606                                      ;LIST OF PATTERNS USED IN WRITING
  5607
  5608  024324  024344                      PATLST: PAT0
  5609  024326  024404                              PAT1
  5610  024330  024444                              PAT2
  5611  024332  024504                              PAT3
  5612  024334  024544                              PAT4
  5613  024336  024604                              PAT5
  5614  024340  024644                              PAT6
  5615  024342  024704                              PAT7
  5616
  5617  024344  000000                      PAT0:   .WORD   0
  5618  024346  000000                              .WORD   0
  5619  024350  000000                              .WORD   0
  5620  024352  000000                              .WORD   0
  5621  024354  000000                              .WORD   0
  5622  024356  000000                              .WORD   0
  5623  024360  000000                              .WORD   0
  5624  024362  000000                              .WORD   0
  5625  024364  000000                              .WORD   0
  5626  024366  000000                              .WORD   0
  5627  024370  000000                              .WORD   0
  5628  024372  000000                              .WORD   0
  5629  024374  000000                              .WORD   0
  5630  024376  000000                              .WORD   0
  5631  024400  000000                              .WORD   0
  5632  024402  000000                              .WORD   0
  5633
  5634  024404  177777                      PAT1:   .WORD   177777
  5635  024406  177777                              .WORD   177777
```

```
024410  177777                  .WORD   177777
024412  052525                  .WORD   052525
024414  052525                  .WORD   052525
024416  052525                  .WORD   052525
024420  177777                  .WORD   177777
024422  177777                  .WORD   177777
024424  052525                  .WORD   052525
024426  052525                  .WORD   052525
024430  177777                  .WORD   177777
024432  052525                  .WORD   052525
024434  177252                  .WORD   177252
024436  177252                  .WORD   177252
024440  172765                  .WORD   172765
024442  172765                  .WORD   172765

024444  000000          PAT2:   .WORD   0
024446  000000                  .WORD   0
024450  000000                  .WORD   0
024452  177777                  .WORD   177777
024454  177777                  .WORD   177777
024456  177777                  .WORD   177777
024460  000000                  .WORD   0
024462  000000                  .WORD   0
024464  177777                  .WORD   177777
024466  177777                  .WORD   177777
024470  000000                  .WORD   0
024472  177777                  .WORD   177777
024474  000000                  .WORD   0
024476  177777                  .WORD   177777
024500  000000                  .WORD   0
024502  177777                  .WORD   177777

024504  025252          PAT3:   .WORD   25252
024506  052525                  .WORD   52525
024510  052525                  .WORD   52525
024512  125252                  .WORD   125252
024514  125252                  .WORD   125252
024516  125252                  .WORD   125252
024520  052525                  .WORD   52525
024522  052525                  .WORD   52525
024524  125252                  .WORD   125252
024526  125252                  .WORD   125252
024530  052525                  .WORD   52525
024532  125252                  .WORD   125252
024534  052525                  .WORD   52525
024536  125252                  .WORD   125252
024540  052525                  .WORD   52525
024542  125252                  .WORD   125252

024544  155555          PAT4:   .WORD   155555
024546  133333                  .WORD   133333
024550  066666                  .WORD   066666
024552  155555                  .WORD   155555
024554  133333                  .WORD   133333
024556  066666                  .WORD   066666
024560  155555                  .WORD   155555
```

```
  5692  024562  133333                          .WORD   133333
  5693  024564  066666                          .WORD   066666
  5694  024566  155555                          .WORD   155555
  5695  024570  133333                          .WORD   133333
  5696  024572  066666                          .WORD   066666
  5697  024574  155555                          .WORD   155555
  5698  024576  133333                          .WORD   133333
  5699  024600  066666                          .WORD   066666
  5700  024602  155555                          .WORD   155555
  5701
  5702  024604  121105                  PAT5:   .WORD   121105
  5703  024606  150442                          .WORD   150442
  5704  024610  064221                          .WORD   64221
  5705  024612  132110                          .WORD   132110
  5706  024614  055044                          .WORD   55044
  5707  024616  026422                          .WORD   26422
  5708  024620  013211                          .WORD   13211
  5709  024622  105504                          .WORD   105504
  5710  024624  042642                          .WORD   42642
  5711  024626  021321                          .WORD   21321
  5712  024630  110550                          .WORD   110550
  5713  024632  044264                          .WORD   44264
  5714  024634  022132                          .WORD   22132
  5715  024636  011055                          .WORD   11055
  5716  024640  104426                          .WORD   104426
  5717  024642  042213                          .WORD   42213
  5718
  5719  024644  177777                  PAT6:   .WORD   177777
  5720  024646  177777                          .WORD   177777
  5721  024650  177777                          .WORD   177777
  5722  024652  177777                          .WORD   177777
  5723  024654  177777                          .WORD   177777
  5724  024656  177777                          .WORD   177777
  5725  024660  177777                          .WORD   177777
  5726  024662  177777                          .WORD   177777
  5727  024664  177777                          .WORD   177777
  5728  024666  177777                          .WORD   177777
  5729  024670  177777                          .WORD   177777
  5730  024672  177777                          .WORD   177777
  5731  024674  177777                          .WORD   177777
  5732  024676  177777                          .WORD   177777
  5733  024700  177777                          .WORD   177777
  5734  024702  177777                          .WORD   177777
  5735
  5736  024704  045513                  PAT7:   .WORD   45513
  5737  024706  122645                          .WORD   122645
  5738  024710  151322                          .WORD   151322
  5739  024712  064551                          .WORD   64551
  5740  024714  132264                          .WORD   132264
  5741  024716  055132                          .WORD   55132
  5742  024720  026455                          .WORD   26455
  5743  024722  113226                          .WORD   113226
  5744  024724  045513                          .WORD   45513
  5745  024726  122645                          .WORD   122645
  5746  024730  151322                          .WORD   151322
  5747  024732  064551                          .WORD   64551
```

```
 5748  024734  132264                                .WORD   132264
 5749  024736  055132                                .WORD   55132
 5750  024740  026455                                .WORD   26455
 5751  024742  113226                                .WORD   113226
 5752
 5753
 5754
 5755  024744  000240                        ENDOFPROGRAM:   NOP
 5756  024746                                ENDTST
  (3)  024746                                L10022:
  (3)  024746  104001                                EMT     C$ETST
 5757  024750  000000                                HALT
 5758
 5759                                        .SBTTL  DRIVE INFORMATION BUFFERS
 5760
 5761                                        ;DRIVE INFORMATION BUFFER
 5762
 5763
 5764                                        .LIST   ME
 5765
 5766  024752                                DRBUF:
 5811
  (1)  024752  000000                                SKCNT                           ;SEEK OPERATION COUNT
  (1)  024754  000002                                RXFR1                           ;READ OPERATION COUNT (BITS) LOW ORDER
  (1)  024756  000004                                RXFR2                           ; "       "        "      "   HIGH ORDER
  (1)  024760  000006                                WXFR1                           ;WRITE OPERATION COUNT (BITS) LOW ORDER
  (1)  024762  000010                                WXFR2                           ; "       "        "      "   HIGH ORDER
  (1)  024764  000012                                ERRCNT                          ;ERROR COUNT - HARD
  (1)  024766  000014                                SFTCNT                          ;ERROR COUNT - SOFT
  (1)  024770  000016                                SKECNT                          ;SEEK ERROR COUNT
  (1)  024772  000020                                DERCNT                          ;DRIVE ERROR COUNT
  (1)  024774  000022                                DCRCER                          ;DATA CRC ERROR COUNT
  (1)  024776  000024                                HCRCER                          ;HEADER CRC ERROR COUNT
  (1)  025000  000026                                DLTCNT                          ;DATA LATE ERROR COUNT
  (1)  025002  000030                                OPICNT                          ;OPERATION INCOMPLETE ERROR COUNT
  (1)  025004  000032                                HNFERR                          ;HEADER NOT FOUND ERROR COUNT
  (1)  025006  000034                                NXMCNT                          ;NON EXISTANT MEMORY ERROR COUNT
  (1)  025010  000036                                RETRY                           ;PRESENT RETRY NUMBER
  (1)  025012  000040                                BDA                             ;    "     DISK ADDRESS CONTENTS
  (1)  025014  000042                                BMP                             ;PRESENT MULTIPURPOSE CONTENTS
  (1)  025016  000044                                FUNC                            ;LAST FUNCTION LOADED
  (1)  025020  000046                                BCSADR                          ;CSR IMAGE OF LAST COMMAND
  (1)  025022  000050                                LSTHDR                          ;LAST POSITION ON DISK
  (1)  025024  000052                                RTYPE                           ;ERROR ON WHICH RECOVERY IS IN PROGRESS
  (1)  025026  000054                                SKCNT1                          ;SEEK COUNT LOW ORDER
  (1)  025030  000056                                PRFLGS                          ;PROGRAM INTERNAL FLAGS
  (1)  025032  000060                                RXFR3                           ;READ COUNT THIRD
  (1)  025034  000062                                WXFR3                           ;WRITE COUNT THIRD
  (1)  025036  000064                                LSTDA                           ;DISK ADDRESS OF SOFT ERROR
  (1)  025040  000066                                DIFWD                           ;LAST DIFFERENCE WORD OF SEEK
  (1)  025042  000070                                DPHOUR                          ;TIME DRIVE WAS DROPPED
  (1)  025044  000072                                TRERR                           ;TRACKING ERROR COUNT
  (1)  025046  000074                                DATCER
  (1)  025050  000076                                DOWCK                           ;WRITE CHECK NECESARY
  (1)  025052  000100                                SERNM1                          ;SERIAL NUMBER OF CARTRIDGE
  (1)  025054  000102                                SERNM2                          ;SERIAL NUMBER OF CARTRIDGE
```

```
   (1)  025056  000104                     DCS                         ;CSR ADDRESS
   (1)  025060  000106                     DRSEL                       ;DRIVE SELECT BITS(8,9,10)
   (1)  025062  000110                     BBA                         ;PRESENT BUS ADDRESS CONTENTS
   (1)  025064  000112                     BSECPT                      ;POINTER TO BAD SECTOR FILE
   (1)  025066  000114                     RSEEK
   (1)  025070  000116                     SOFTCS                      ;CSR AT TIME OF SOFT ERROR
   (1)  025072  000120                     PRPOS                       ;PRESENT POSITION ON DISK
   (1)
   (1)
   (1)  025074  000000                     SKCNT                       ;SEEK OPERATION COUNT
   (1)  025076  000002                     RXFR1                       ;READ OPERATION COUNT (BITS) LOW ORDER
   (1)  025100  000004                     RXFR2                       ; "      "      "     "   HIGH ORDER
   (1)  025102  000006                     WXFR1                       ;WRITE OPERATION COUNT (BITS) LOW ORDER
   (1)  025104  000010                     WXFR2                       ; "      "      "     "   HIGH ORDER
   (1)  025106  000012                     ERRCNT                      ;ERROR COUNT - HARD
   (1)  025110  000014                     SFTCNT                      ;ERROR COUNT - SOFT
   (1)  025112  000016                     SKECNT                      ;SEEK ERROR COUNT
   (1)  025114  000020                     DERCNT                      ;DRIVE ERROR COUNT
   (1)  025116  000022                     DCRCER                      ;DATA CRC ERROR COUNT
   (1)  025120  000024                     HCRCER                      ;HEADER CRC ERROR COUNT
   (1)  025122  000026                     DLTCNT                      ;DATA LATE ERROR COUNT
   (1)  025124  000030                     OPICNT                      ;OPERATION INCOMPLETE ERROR COUNT
   (1)  025126  000032                     HNFERR                      ;HEADER NOT FOUND ERROR COUNT
   (1)  025130  000034                     NXMCNT                      ;NON EXISTANT MEMORY ERROR COUNT
   (1)  025132  000036                     RETRY                       ;PRESENT RETRY NUMBER
   (1)  025134  000040                     BDA                         ;    "     DISK ADDRESS CONTENTS
   (1)  025136  000042                     BMP                         ;PRESENT MULTIPURPOSE CONTENTS
   (1)  025140  000044                     FUNC                        ;LAST FUNCTION LOADED
   (1)  025142  000046                     BCSADR                      ;CSR IMAGE OF LAST COMMAND
   (1)  025144  000050                     LSTHDR                      ;LAST POSITION ON DISK
   (1)  025146  000052                     RTYPE                       ;ERROR ON WHICH RECOVERY IS IN PROGRESS
   (1)  025150  000054                     SKCNT1                      ;SEEK COUNT LOW ORDER
   (1)  025152  000056                     PRFLGS                      ;PROGRAM INTERNAL FLAGS
   (1)  025154  000060                     RXFR3                       ;READ COUNT THIRD
   (1)  025156  000062                     WXFR3                       ;WRITE COUNT THIRD
   (1)  025160  000064                     LSTDA                       ;DISK ADDRESS OF SOFT ERROR
   (1)  025162  000066                     DIFWD                       ;LAST DIFFERENCE WORD OF SEEK
   (1)  025164  000070                     DPHOUR                      ;TIME DRIVE WAS DROPPED
   (1)  025166  000072                     TRERR                       ;TRACKING ERROR COUNT
   (1)  025170  000074                     DATCER
   (1)  025172  000076                     DOWCK                       ;WRITE CHECK NECESARY
   (1)  025174  000100                     SERNM1                      ;SERIAL NUMBER OF CARTRIDGE
   (1)  025176  000102                     SERNM2                      ;SERIAL NUMBER OF CARTRIDGE
   (1)  025200  000104                     DCS                         ;CSR ADDRESS
   (1)  025202  000106                     DRSEL                       ;DRIVE SELECT BITS(8,9,10)
   (1)  025204  000110                     BBA                         ;PRESENT BUS ADDRESS CONTENTS
   (1)  025206  000112                     BSECPT                      ;POINTER TO BAD SECTOR FILE
   (1)  025210  000114                     RSEEK
   (1)  025212  000116                     SOFTCS                      ;CSR AT TIME OF SOFT ERROR
   (1)  025214  000120                     PRPOS                       ;PRESENT POSITION ON DISK
   (1)
   (1)
   (1)  025216  000000                     SKCNT                       ;SEEK OPERATION COUNT
   (1)  025220  000002                     RXFR1                       ;READ OPERATION COUNT (BITS) LOW ORDER
   (1)  025222  000004                     RXFR2                       ; "      "      "     "   HIGH ORDER
   (1)  025224  000006                     WXFR1                       ;WRITE OPERATION COUNT (BITS) LOW ORDER
```

```
  (1)  025226  000010                          WXFR2                        ;  "        "        "        "   HIGH ORDER
  (1)  025230  000012                          ERRCNT                       ;ERROR COUNT - HARD
  (1)  025232  000014                          SFTCNT                       ;ERROR COUNT - SOFT
  (1)  025234  000016                          SKECNT                       ;SEEK ERROR COUNT
  (1)  025236  000020                          DERCNT                       ;DRIVE ERROR COUNT
  (1)  025240  000022                          DCRCER                       ;DATA CRC ERROR COUNT
  (1)  025242  000024                          HCRCER                       ;HEADER CRC ERROR COUNT
  (1)  025244  000026                          DLTCNT                       ;DATA LATE ERROR COUNT
  (1)  025246  000030                          OPICNT                       ;OPERATION INCOMPLETE ERROR COUNT
  (1)  025250  000032                          HNFERR                       ;HEADER NOT FOUND ERROR COUNT
  (1)  025252  000034                          NXMCNT                       ;NON EXISTANT MEMORY ERROR COUNT
  (1)  025254  000036                          RETRY                        ;PRESENT RETRY NUMBER
  (1)  025256  000040                          BDA                          ;    "     DISK ADDRESS CONTENTS
  (1)  025260  000042                          BMP                          ;PRESENT MULTIPURPOSE CONTENTS
  (1)  025262  000044                          FUNC                         ;LAST FUNCTION LOADED
  (1)  025264  000046                          BCSADR                       ;CSR IMAGE OF LAST COMMAND
  (1)  025266  000050                          LSTHDR                       ;LAST POSITION ON DISK
  (1)  025270  000052                          RTYPE                        ;ERROR ON WHICH RECOVERY IS IN PROGRESS
  (1)  025272  000054                          SKCNT1                       ;SEEK COUNT LOW ORDER
  (1)  025274  000056                          PRFLGS                       ;PROGRAM INTERNAL FLAGS
  (1)  025276  000060                          RXFR3                        ;READ COUNT THIRD
  (1)  025300  000062                          WXFR3                        ;WRITE COUNT THIRD
  (1)  025302  000064                          LSTDA                        ;DISK ADDRESS OF SOFT ERROR
  (1)  025304  000066                          DIFWD                        ;LAST DIFFERENCE WORD OF SEEK
  (1)  025306  000070                          DPHOUR                       ;TIME DRIVE WAS DROPPED
  (1)  025310  000072                          TRERR                        ;TRACKING ERROR COUNT
  (1)  025312  000074                          DATCER
  (1)  025314  000076                          DOWCK                        ;WRITE CHECK NECESARY
  (1)  025316  000100                          SERNM1                       ;SERIAL NUMBER OF CARTRIDGE
  (1)  025320  000102                          SERNM2                       ;SERIAL NUMBER OF CARTRIDGE
  (1)  025322  000104                          DCS                          ;CSR ADDRESS
  (1)  025324  000106                          DRSEL                        ;DRIVE SELECT BITS(8,9,10)
  (1)  025326  000110                          BBA                          ;PRESENT BUS ADDRESS CONTENTS
  (1)  025330  000112                          BSECPT                       ;POINTER TO BAD SECTOR FILE
  (1)  025332  000114                          RSEEK
  (1)  025334  000116                          SOFTCS                       ;CSR AT TIME OF SOFT ERROR
  (1)  025336  000120                          PRPOS                        ;PRESENT POSITION ON DISK
  (1)
  (1)
  (1)  025340  000000                          SKCNT                        ;SEEK OPERATION COUNT
  (1)  025342  000002                          RXFR1                        ;READ OPERATION COUNT (BITS) LOW ORDER
  (1)  025344  000004                          RXFR2                        ;  "        "        "        "   HIGH ORDER
  (1)  025346  000006                          WXFR1                        ;WRITE OPERATION COUNT (BITS) LOW ORDER
  (1)  025350  000010                          WXFR2                        ;  "        "        "        "   HIGH ORDER
  (1)  025352  000012                          ERRCNT                       ;ERROR COUNT - HARD
  (1)  025354  000014                          SFTCNT                       ;ERROR COUNT - SOFT
  (1)  025356  000016                          SKECNT                       ;SEEK ERROR COUNT
  (1)  025360  000020                          DERCNT                       ;DRIVE ERROR COUNT
  (1)  025362  000022                          DCRCER                       ;DATA CRC ERROR COUNT
  (1)  025364  000024                          HCRCER                       ;HEADER CRC ERROR COUNT
  (1)  025366  000026                          DLTCNT                       ;DATA LATE ERROR COUNT
  (1)  025370  000030                          OPICNT                       ;OPERATION INCOMPLETE ERROR COUNT
  (1)  025372  000032                          HNFERR                       ;HEADER NOT FOUND ERROR COUNT
  (1)  025374  000034                          NXMCNT                       ;NON EXISTANT MEMORY ERROR COUNT
  (1)  025376  000036                          RETRY                        ;PRESENT RETRY NUMBER
  (1)  025400  000040                          BDA                          ;    "     DISK ADDRESS CONTENTS
```

```
    (1)   025402  000042                          BMP                       ;PRESENT MULTIPURPOSE CONTENTS
    (1)   025404  000044                          FUNC                      ;LAST FUNCTION LOADED
    (1)   025406  000046                          BCSADR                    ;CSR IMAGE OF LAST COMMAND
    (1)   025410  000050                          LSTHDR                    ;LAST POSITION ON DISK
    (1)   025412  000052                          RTYPE                     ;ERROR ON WHICH RECOVERY IS IN PROGRESS
    (1)   025414  000054                          SKCNT1                    ;SEEK COUNT LOW ORDER
    (1)   025416  000056                          PRFLGS                    ;PROGRAM INTERNAL FLAGS
    (1)   025420  000060                          RXFR3                     ;READ COUNT THIRD
    (1)   025422  000062                          WXFR3                     ;WRITE COUNT THIRD
    (1)   025424  000064                          LSTDA                     ;DISK ADDRESS OF SOFT ERROR
    (1)   025426  000066                          DIFWD                     ;LAST DIFFERENCE WORD OF SEEK
    (1)   025430  000070                          DPHOUR                    ;TIME DRIVE WAS DROPPED
    (1)   025432  000072                          TRERR                     ;TRACKING ERROR COUNT
    (1)   025434  000074                          DATCER
    (1)   025436  000076                          DOWCK                     ;WRITE CHECK NECESARY
    (1)   025440  000100                          SERNM1                    ;SERIAL NUMBER OF CARTRIDGE
    (1)   025442  000102                          SERNM2                    ;SERIAL NUMBER OF CARTRIDGE
    (1)   025444  000104                          DCS                       ;CSR ADDRESS
    (1)   025446  000106                          DRSEL                     ;DRIVE SELECT BITS(8,9,10)
    (1)   025450  000110                          BBA                       ;PRESENT BUS ADDRESS CONTENTS
    (1)   025452  000112                          BSECPT                    ;POINTER TO BAD SECTOR FILE
    (1)   025454  000114                          RSEEK
    (1)   025456  000116                          SOFTCS                    ;CSR AT TIME OF SOFT ERROR
    (1)   025460  000120                          PRPOS                     ;PRESENT POSITION ON DISK
    (1)
    (1)
    (1)   025462  000000                          SKCNT                     ;SEEK OPERATION COUNT
    (1)   025464  000002                          RXFR1                     ;READ OPERATION COUNT (BITS) LOW ORDER
    (1)   025466  000004                          RXFR2                     ; "        "        "       "   HIGH ORDER
    (1)   025470  000006                          WXFR1                     ;WRITE OPERATION COUNT (BITS) LOW ORDER
    (1)   025472  000010                          WXFR2                     ; "        "        "       "   HIGH ORDER
    (1)   025474  000012                          ERRCNT                    ;ERROR COUNT - HARD
    (1)   025476  000014                          SFTCNT                    ;ERROR COUNT - SOFT
    (1)   025500  000016                          SKECNT                    ;SEEK ERROR COUNT
    (1)   025502  000020                          DERCNT                    ;DRIVE ERROR COUNT
    (1)   025504  000022                          DCRCER                    ;DATA CRC ERROR COUNT
    (1)   025506  000024                          HCRCER                    ;HEADER CRC ERROR COUNT
    (1)   025510  000026                          DLTCNT                    ;DATA LATE ERROR COUNT
    (1)   025512  000030                          OPICNT                    ;OPERATION INCOMPLETE ERROR COUNT
    (1)   025514  000032                          HNFERR                    ;HEADER NOT FOUND ERROR COUNT
    (1)   025516  000034                          NXMCNT                    ;NON EXISTANT MEMORY ERROR COUNT
    (1)   025520  000036                          RETRY                     ;PRESENT RETRY NUMBER
    (1)   025522  000040                          BDA                       ;    "     DISK ADDRESS CONTENTS
    (1)   025524  000042                          BMP                       ;PRESENT MULTIPURPOSE CONTENTS
    (1)   025526  000044                          FUNC                      ;LAST FUNCTION LOADED
    (1)   025530  000046                          BCSADR                    ;CSR IMAGE OF LAST COMMAND
    (1)   025532  000050                          LSTHDR                    ;LAST POSITION ON DISK
    (1)   025534  000052                          RTYPE                     ;ERROR ON WHICH RECOVERY IS IN PROGRESS
    (1)   025536  000054                          SKCNT1                    ;SEEK COUNT LOW ORDER
    (1)   025540  000056                          PRFLGS                    ;PROGRAM INTERNAL FLAGS
    (1)   025542  000060                          RXFR3                     ;READ COUNT THIRD
    (1)   025544  000062                          WXFR3                     ;WRITE COUNT THIRD
    (1)   025546  000064                          LSTDA                     ;DISK ADDRESS OF SOFT ERROR
    (1)   025550  000066                          DIFWD                     ;LAST DIFFERENCE WORD OF SEEK
    (1)   025552  000070                          DPHOUR                    ;TIME DRIVE WAS DROPPED
    (1)   025554  000072                          TRERR                     ;TRACKING ERROR COUNT
    (1)  025556  000074                        DATCER
    (1)  025560  000076                        DOWCK                   ;WRITE CHECK NECESARY
    (1)  025562  000100                        SERNM1                  ;SERIAL NUMBER OF CARTRIDGE
    (1)  025564  000102                        SERNM2                  ;SERIAL NUMBER OF CARTRIDGE
    (1)  025566  000104                        DCS                     ;CSR ADDRESS
    (1)  025570  000106                        DRSEL                   ;DRIVE SELECT BITS(8,9,10)
    (1)  025572  000110                        BBA                     ;PRESENT BUS ADDRESS CONTENTS
    (1)  025574  000112                        BSECPT                  ;POINTER TO BAD SECTOR FILE
    (1)  025576  000114                        RSEEK
    (1)  025600  000116                        SOFTCS                  ;CSR AT TIME OF SOFT ERROR
    (1)  025602  000120                        PRPOS                   ;PRESENT POSITION ON DISK
    (1)
    (1)
    (1)  025604  000000                        SKCNT                   ;SEEK OPERATION COUNT
    (1)  025606  000002                        RXFR1                   ;READ OPERATION COUNT (BITS) LOW ORDER
    (1)  025610  000004                        RXFR2                   ;  "       "       "      "  HIGH ORDER
    (1)  025612  000006                        WXFR1                   ;WRITE OPERATION COUNT (BITS) LOW ORDER
    (1)  025614  000010                        WXFR2                   ;  "       "       "      "  HIGH ORDER
    (1)  025616  000012                        ERRCNT                  ;ERROR COUNT - HARD
    (1)  025620  000014                        SFTCNT                  ;ERROR COUNT - SOFT
    (1)  025622  000016                        SKECNT                  ;SEEK ERROR COUNT
    (1)  025624  000020                        DERCNT                  ;DRIVE ERROR COUNT
    (1)  025626  000022                        DCRCER                  ;DATA CRC ERROR COUNT
    (1)  025630  000024                        HCRCER                  ;HEADER CRC ERROR COUNT
    (1)  025632  000026                        DLTCNT                  ;DATA LATE ERROR COUNT
    (1)  025634  000030                        OPICNT                  ;OPERATION INCOMPLETE ERROR COUNT
    (1)  025636  000032                        HNFERR                  ;HEADER NOT FOUND ERROR COUNT
    (1)  025640  000034                        NXMCNT                  ;NON EXISTANT MEMORY ERROR COUNT
    (1)  025642  000036                        RETRY                   ;PRESENT RETRY NUMBER
    (1)  025644  000040                        BDA                     ;   "    DISK ADDRESS CONTENTS
    (1)  025646  000042                        BMP                     ;PRESENT MULTIPURPOSE CONTENTS
    (1)  025650  000044                        FUNC                    ;LAST FUNCTION LOADED
    (1)  025652  000046                        BCSADR                  ;CSR IMAGE OF LAST COMMAND
    (1)  025654  000050                        LSTHDR                  ;LAST POSITION ON DISK
    (1)  025656  000052                        RTYPE                   ;ERROR ON WHICH RECOVERY IS IN PROGRESS
    (1)  025660  000054                        SKCNT1                  ;SEEK COUNT LOW ORDER
    (1)  025662  000056                        PRFLGS                  ;PROGRAM INTERNAL FLAGS
    (1)  025664  000060                        RXFR3                   ;READ COUNT THIRD
    (1)  025666  000062                        WXFR3                   ;WRITE COUNT THIRD
    (1)  025670  000064                        LSTDA                   ;DISK ADDRESS OF SOFT ERROR
    (1)  025672  000066                        DIFWD                   ;LAST DIFFERENCE WORD OF SEEK
    (1)  025674  000070                        DPHOUR                  ;TIME DRIVE WAS DROPPED
    (1)  025676  000072                        TRERR                   ;TRACKING ERROR COUNT
    (1)  025700  000074                        DATCER
    (1)  025702  000076                        DOWCK                   ;WRITE CHECK NECESARY
    (1)  025704  000100                        SERNM1                  ;SERIAL NUMBER OF CARTRIDGE
    (1)  025706  000102                        SERNM2                  ;SERIAL NUMBER OF CARTRIDGE
    (1)  025710  000104                        DCS                     ;CSR ADDRESS
    (1)  025712  000106                        DRSEL                   ;DRIVE SELECT BITS(8,9,10)
    (1)  025714  000110                        BBA                     ;PRESENT BUS ADDRESS CONTENTS
    (1)  025716  000112                        BSECPT                  ;POINTER TO BAD SECTOR FILE
    (1)  025720  000114                        RSEEK
    (1)  025722  000116                        SOFTCS                  ;CSR AT TIME OF SOFT ERROR
    (1)  025724  000120                        PRPOS                   ;PRESENT POSITION ON DISK
    (1)
    (1)
    (1)  025726  000000                     SKCNT                  ;SEEK OPERATION COUNT
    (1)  025730  000002                     RXFR1                  ;READ OPERATION COUNT (BITS) LOW ORDER
    (1)  025732  000004                     RXFR2                  ;  "       "        "       "    HIGH ORDER
    (1)  025734  000006                     WXFR1                  ;WRITE OPERATION COUNT (BITS) LOW ORDER
    (1)  025736  000010                     WXFR2                  ;  "       "        "       "    HIGH ORDER
    (1)  025740  000012                     ERRCNT                 ;ERROR COUNT - HARD
    (1)  025742  000014                     SFTCNT                 ;ERROR COUNT - SOFT
    (1)  025744  000016                     SKECNT                 ;SEEK ERROR COUNT
    (1)  025746  000020                     DERCNT                 ;DRIVE ERROR COUNT
    (1)  025750  000022                     DCRCER                 ;DATA CRC ERROR COUNT
    (1)  025752  000024                     HCRCER                 ;HEADER CRC ERROR COUNT
    (1)  025754  000026                     DLTCNT                 ;DATA LATE ERROR COUNT
    (1)  025756  000030                     OPICNT                 ;OPERATION INCOMPLETE ERROR COUNT
    (1)  025760  000032                     HNFERR                 ;HEADER NOT FOUND ERROR COUNT
    (1)  025762  000034                     NXMCNT                 ;NON EXISTANT MEMORY ERROR COUNT
    (1)  025764  000036                     RETRY                  ;PRESENT RETRY NUMBER
    (1)  025766  000040                     BDA                    ;    "     DISK ADDRESS CONTENTS
    (1)  025770  000042                     BMP                    ;PRESENT MULTIPURPOSE CONTENTS
    (1)  025772  000044                     FUNC                   ;LAST FUNCTION LOADED
    (1)  025774  000046                     BCSADR                 ;CSR IMAGE OF LAST COMMAND
    (1)  025776  000050                     LSTHDR                 ;LAST POSITION ON DISK
    (1)  026000  000052                     RTYPE                  ;ERROR ON WHICH RECOVERY IS IN PROGRESS
    (1)  026002  000054                     SKCNT1                 ;SEEK COUNT LOW ORDER
    (1)  026004  000056                     PRFLGS                 ;PROGRAM INTERNAL FLAGS
    (1)  026006  000060                     RXFR3                  ;READ COUNT THIRD
    (1)  026010  000062                     WXFR3                  ;WRITE COUNT THIRD
    (1)  026012  000064                     LSTDA                  ;DISK ADDRESS OF SOFT ERROR
    (1)  026014  000066                     DIFWD                  ;LAST DIFFERENCE WORD OF SEEK
    (1)  026016  000070                     DPHOUR                 ;TIME DRIVE WAS DROPPED
    (1)  026020  000072                     TRERR                  ;TRACKING ERROR COUNT
    (1)  026022  000074                     DATCER
    (1)  026024  000076                     DOWCK                  ;WRITE CHECK NECESARY
    (1)  026026  000100                     SERNM1                 ;SERIAL NUMBER OF CARTRIDGE
    (1)  026030  000102                     SERNM2                 ;SERIAL NUMBER OF CARTRIDGE
    (1)  026032  000104                     DCS                    ;CSR ADDRESS
    (1)  026034  000106                     DRSEL                  ;DRIVE SELECT BITS(8,9,10)
    (1)  026036  000110                     BBA                    ;PRESENT BUS ADDRESS CONTENTS
    (1)  026040  000112                     BSECPT                 ;POINTER TO BAD SECTOR FILE
    (1)  026042  000114                     RSEEK
    (1)  026044  000116                     SOFTCS                 ;CSR AT TIME OF SOFT ERROR
    (1)  026046  000120                     PRPOS                  ;PRESENT POSITION ON DISK
    (1)
    (1)
    (1)  026050  000000                     SKCNT                  ;SEEK OPERATION COUNT
    (1)  026052  000002                     RXFR1                  ;READ OPERATION COUNT (BITS) LOW ORDER
    (1)  026054  000004                     RXFR2                  ;  "       "        "       "    HIGH ORDER
    (1)  026056  000006                     WXFR1                  ;WRITE OPERATION COUNT (BITS) LOW ORDER
    (1)  026060  000010                     WXFR2                  ;  "       "        "       "    HIGH ORDER
    (1)  026062  000012                     ERRCNT                 ;ERROR COUNT - HARD
    (1)  026064  000014                     SFTCNT                 ;ERROR COUNT - SOFT
    (1)  026066  000016                     SKECNT                 ;SEEK ERROR COUNT
    (1)  026070  000020                     DERCNT                 ;DRIVE ERROR COUNT
    (1)  026072  000022                     DCRCER                 ;DATA CRC ERROR COUNT
    (1)  026074  000024                     HCRCER                 ;HEADER CRC ERROR COUNT
    (1)  026076  000026                     DLTCNT                 ;DATA LATE ERROR COUNT
    (1)  026100  000030                     OPICNT                 ;OPERATION INCOMPLETE ERROR COUNT
```

```
   (1)  026102  000032                          HNFERR                  ;HEADER NOT FOUND ERROR COUNT
   (1)  026104  000034                          NXMCNT                  ;NON EXISTANT MEMORY ERROR COUNT
   (1)  026106  000036                          RETRY                   ;PRESENT RETRY NUMBER
   (1)  026110  000040                          BDA                     ;    "     DISK ADDRESS CONTENTS
   (1)  026112  000042                          BMP                     ;PRESENT MULTIPURPOSE CONTENTS
   (1)  026114  000044                          FUNC                    ;LAST FUNCTION LOADED
   (1)  026116  000046                          BCSADR                  ;CSR IMAGE OF LAST COMMAND
   (1)  026120  000050                          LSTHDR                  ;LAST POSITION ON DISK
   (1)  026122  000052                          RTYPE                   ;ERROR ON WHICH RECOVERY IS IN PROGRESS
   (1)  026124  000054                          SKCNT1                  ;SEEK COUNT LOW ORDER
   (1)  026126  000056                          PRFLGS                  ;PROGRAM INTERNAL FLAGS
   (1)  026130  000060                          RXFR3                   ;READ COUNT THIRD
   (1)  026132  000062                          WXFR3                   ;WRITE COUNT THIRD
   (1)  026134  000064                          LSTDA                   ;DISK ADDRESS OF SOFT ERROR
   (1)  026136  000066                          DIFWD                   ;LAST DIFFERENCE WORD OF SEEK
   (1)  026140  000070                          DPHOUR                  ;TIME DRIVE WAS DROPPED
   (1)  026142  000072                          TRERR                   ;TRACKING ERROR COUNT
   (1)  026144  000074                          DATCER
   (1)  026146  000076                          DOWCK                   ;WRITE CHECK NECESARY
   (1)  026150  000100                          SERNM1                  ;SERIAL NUMBER OF CARTRIDGE
   (1)  026152  000102                          SERNM2                  ;SERIAL NUMBER OF CARTRIDGE
   (1)  026154  000104                          DCS                     ;CSR ADDRESS
   (1)  026156  000106                          DRSEL                   ;DRIVE SELECT BITS(8,9,10)
   (1)  026160  000110                          BBA                     ;PRESENT BUS ADDRESS CONTENTS
   (1)  026162  000112                          BSECPT                  ;POINTER TO BAD SECTOR FILE
   (1)  026164  000114                          RSEEK
   (1)  026166  000116                          SOFTCS                  ;CSR AT TIME OF SOFT ERROR
   (1)  026170  000120                          PRPOS                   ;PRESENT POSITION ON DISK
   (1)
  5812                                  .NLIST  ME
  5813
  5814  026172  000000                  ENDBUF: .WORD   0
  5815
  5816
  5817  026174                          BGNMOD  HRDPRM
  5818
  5819  026174                                  BGNHRD
   (3)  026174  000025                          .WORD L10024-L$HARD/2
  5820
  5821  026176                                  GPRML   CNTYPE,CNT,1,YES
   (4)  026176  004130                          .WORD   T$CODE
   (4)  026200  026250                          .WORD   CNTYPE
   (4)  026202  000001                          .WORD   1
  5822  026204                                  GPRMA   CSRMSG,CSR,0,160000,177776,YES
   (4)  026204  000031                          .WORD   T$CODE
   (4)  026206  026255                          .WORD   CSRMSG
   (4)  026210  160000                          .WORD   T$LOLIM
   (4)  026212  177776                          .WORD   T$HILIM
  5823  026214                                  GPRMA   VECMSG,VECT,0,0,776,YES
   (4)  026214  001031                          .WORD   T$CODE
   (4)  026216  026302                          .WORD   VECMSG
   (4)  026220  000000                          .WORD   T$LOLIM
   (4)  026222  000776                          .WORD   T$HILIM
  5824  026224                                  GPRMD   BRMSG,PRIOR,0,340,0,7,YES
   (4)  026224  002032                          .WORD   T$CODE
   (4)  026226  026271                          .WORD   BRMSG
  (4)  026230  000340                             .WORD   340
  (4)  026232  000000                             .WORD   T$LOLIM
  (4)  026234  000007                             .WORD   T$HILIM
 5825  026236                                     GPRMD   DRMSG,DRBT,0,03400,0,7,YES
  (4)  026236  003032                             .WORD   T$CODE
  (4)  026240  026311                             .WORD   DRMSG
  (4)  026242  003400                             .WORD   03400
  (4)  026244  000000                             .WORD   T$LOLIM
  (4)  026246  000007                             .WORD   T$HILIM
 5826
 5827  026250                                     ENDHRD
  (2)                                             .EVEN
  (3)  026250                           L10024:
 5828
 5832
 5833  026250  046122  030461     000  CNTYPE:   .ASCIZ  /RL11/
 5834  026255     102  051525  040440  CSRMSG:   .ASCIZ  /BUS ADDRESS/
 5835  026271     102  020122  042514  BRMSG:    .ASCIZ  /BR LEVEL/
 5836  026302  042526  052103  051117  VECMSG:   .ASCIZ  /VECTOR/
 5837  026311     104  044522  042526  DRMSG:    .ASCIZ  /DRIVE/
 5838
 5842
 5843          026320                   .EVEN
 5844
 5845  026320                                     ENDMOD
 5846
 5847  026320                           BGNMOD    SFTPRM
 5848
 5849  026320                                     BGNSFT
  (3)  026320  000217                             .WORD L10025-L$SOFT/2
 5850
 5851  026322                                     GPRMD   RTMSG,RLT,D,177777,0,177777,YES
  (4)  026322  000052                             .WORD   T$CODE
  (4)  026324  027165                             .WORD   RTMSG
  (4)  026326  177777                             .WORD   177777
  (4)  026330  000000                             .WORD   T$LOLIM
  (4)  026332  177777                             .WORD   T$HILIM
 5852  026334                                     GPRMD   SRTMSG,SRLT,D,177777,0,177777,YES
  (4)  026334  031052                             .WORD   T$CODE
  (4)  026336  027010                             .WORD   SRTMSG
  (4)  026340  177777                             .WORD   177777
  (4)  026342  000000                             .WORD   T$LOLIM
  (4)  026344  177777                             .WORD   T$HILIM
 5853  026346                                     GPRML   FDCHK,DCKFG,1,YES
  (4)  026346  020130                             .WORD   T$CODE
  (4)  026350  027472                             .WORD   FDCHK
  (4)  026352  000001                             .WORD   1
 5854  026354                                     XFERF   5$
  (5)  026354  006044                             .WORD   T$CODE
 5855  026356                                     GPRMD   CHKLMT,CLMT,D,177777,0,128.,YES
  (4)  026356  032052                             .WORD   T$CODE
  (4)  026360  027027                             .WORD   CHKLMT
  (4)  026362  177777                             .WORD   177777
  (4)  026364  000000                             .WORD   T$LOLIM
  (4)  026366  000200                             .WORD   T$HILIM
 5856  026370                           5$:       GPRMD   INMSG,TYT,D,177777,1,177777,YES
```

```
   (4)   026370  005052                        .WORD    T$CODE
   (4)   026372  027275                        .WORD    INMSG
   (4)   026374  177777                        .WORD    177777
   (4)   026376  000001                        .WORD    T$LOLIM
   (4)   026400  177777                        .WORD    T$HILIM
  5857   026402                                GPRML    DRPMS,DRFLG,1,YES
   (4)   026402  021130                        .WORD    T$CODE
   (4)   026404  027553                        .WORD    DRPMS
   (4)   026406  000001                        .WORD    1
  5858   026410                                XFERF    3$
   (5)   026410  032044                        .WORD    T$CODE
  5859   026412                                GPRMD    ERMSG,ELT,D,177777,0,177777,YES
   (4)   026412  001052                        .WORD    T$CODE
   (4)   026414  027101                        .WORD    ERMSG
   (4)   026416  177777                        .WORD    177777
   (4)   026420  000000                        .WORD    T$LOLIM
   (4)   026422  177777                        .WORD    T$HILIM
  5860   026424                                GPRMD    SFTMSG,SEL,D,177777,0,177777,YES
   (4)   026424  023052                        .WORD    T$CODE
   (4)   026426  027115                        .WORD    SFTMSG
   (4)   026430  177777                        .WORD    177777
   (4)   026432  000000                        .WORD    T$LOLIM
   (4)   026434  177777                        .WORD    T$HILIM
  5861   026436                                GPRMD    DERPMS,DCD,D,177777,0,177777,YES
   (4)   026436  036052                        .WORD    T$CODE
   (4)   026440  027616                        .WORD    DERPMS
   (4)   026442  177777                        .WORD    177777
   (4)   026444  000000                        .WORD    T$LOLIM
   (4)   026446  177777                        .WORD    T$HILIM
  5862   026450                                GPRMD    SEMSG,SET,D,177777,0,177777,YES
   (4)   026450  002052                        .WORD    T$CODE
   (4)   026452  027177                        .WORD    SEMSG
   (4)   026454  177777                        .WORD    177777
   (4)   026456  000000                        .WORD    T$LOLIM
   (4)   026460  177777                        .WORD    T$HILIM
  5863   026462                                GPRMD    DREMSG,DET,D,177777,0,177777,YES
   (4)   026462  025052                        .WORD    T$CODE
   (4)   026464  027212                        .WORD    DREMSG
   (4)   026466  177777                        .WORD    177777
   (4)   026470  000000                        .WORD    T$LOLIM
   (4)   026472  177777                        .WORD    T$HILIM
  5864   026474                        3$:     GPRML    STLMT,OPFLG,1,YES
   (4)   026474  024130                        .WORD    T$CODE
   (4)   026476  027516                        .WORD    STLMT
   (4)   026500  000001                        .WORD    1
  5865   026502                                XFERF    2$
   (5)   026502  013044                        .WORD    T$CODE
  5866   026504                                GPRMD    DAMSG,DAT,D,177777,1,177776,YES
   (4)   026504  003052                        .WORD    T$CODE
   (4)   026506  027225                        .WORD    DAMSG
   (4)   026510  177777                        .WORD    177777
   (4)   026512  000001                        .WORD    T$LOLIM
   (4)   026514  177776                        .WORD    T$HILIM
  5867   026516                                GPRMD    SKMSG,SKT,D,177777,1,177776,YES
   (4)   026516  004052                        .WORD    T$CODE
   (4)   026520  027255                        .WORD    SKMSG
   (4)  026522  177777                      .WORD   177777
   (4)  026524  000001                      .WORD   T$LOLIM
   (4)  026526  177776                      .WORD   T$HILIM
  5868  026530                      2$:     GPRML   CHANGE,CHFLG,1,YES
   (4)  026530  010130                      .WORD   T$CODE
   (4)  026532  027325                      .WORD   CHANGE
   (4)  026534  000001                      .WORD   1
  5869  026536                              XFERF   1$
   (5)  026536  106044                      .WORD   T$CODE
  5870  026540                              GPRML   STIPMS,STIP,1,YES
   (4)  026540  034130                      .WORD   T$CODE
   (4)  026542  026760                      .WORD   STIPMS
   (4)  026544  000001                      .WORD   1
  5871  026546                              XFERF   6$
   (5)  026546  013044                      .WORD   T$CODE
  5872  026550                              GPRMD   MXBUF,MXB,D,177777,3,5120.,YES
   (4)  026550  011052                      .WORD   T$CODE
   (4)  026552  027361                      .WORD   MXBUF
   (4)  026554  177777                      .WORD   177777
   (4)  026556  000003                      .WORD   T$LOLIM
   (4)  026560  012000                      .WORD   T$HILIM
  5873  026562                              GPRMD   MINBUF,MNB,D,177777,3.,5120.,YES
   (4)  026562  022052                      .WORD   T$CODE
   (4)  026564  027372                      .WORD   MINBUF
   (4)  026566  177777                      .WORD   177777
   (4)  026570  000003                      .WORD   T$LOLIM
   (4)  026572  012000                      .WORD   T$HILIM
  5874  026574                      6$:     GPRML   RDONLY,ROF,1,YES
   (4)  026574  026130                      .WORD   T$CODE
   (4)  026576  027047                      .WORD   RDONLY
   (4)  026600  000001                      .WORD   1
  5875  026602                              GPRML   RANPAT,RAN,1,YES
   (4)  026602  027130                      .WORD   T$CODE
   (4)  026604  027057                      .WORD   RANPAT
   (4)  026606  000001                      .WORD   1
  5876  026610                              XFERT   4$
   (5)  026610  006024                      .WORD   T$CODE
  5877  026612                              GPRMD   ONLONE,PAT,0,17,0,7,YES
   (4)  026612  030032                      .WORD   T$CODE
   (4)  026614  027067                      .WORD   ONLONE
   (4)  026616  000017                      .WORD   17
   (4)  026620  000000                      .WORD   T$LOLIM
   (4)  026622  000007                      .WORD   T$HILIM
  5878  026624                      4$:     GPRML   WCKMSG,WCK,1,YES
   (4)  026624  035130                      .WORD   T$CODE
   (4)  026626  027607                      .WORD   WCKMSG
   (4)  026630  000001                      .WORD   1
  5879  026632                      7$:     GPRMD   CMMSG,RDT,D,177777,0,128.,YES
   (4)  026632  006052                      .WORD   T$CODE
   (4)  026634  027644                      .WORD   CMMSG
   (4)  026636  177777                      .WORD   177777
   (4)  026640  000000                      .WORD   T$LOLIM
   (4)  026642  000200                      .WORD   T$HILIM
  5880  026644                              GPRMD   DEMSG,DDT,D,177777,0,175,YES
   (4)  026644  007052                      .WORD   T$CODE
   (4)  026646  027131                      .WORD   DEMSG
```