RL11,RLV11

RL01,02 PERF EXER

CZRLKA0

AH-F126A-MC

FICHE 1 OF 1

MAY 1979

digital

MADE IN USA

IDENTIFICATION

PRODUCT CODE: AC-F127A-MC

PRODUCT NAME: CZRLKA0 RL01/02 PERFORMANCE EXERCISER

DATE CREATED: 5-JAN-79

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: D. DEKNIS, C. CAMPBELL

TABLE OF CONTENTS

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 B>). 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 RL11/RLV11 RL01/02 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 RL01/02, THE DRIVES ARE RANDOMLY PICKED AND GIVEN A RANDOM STRING FUNCTION OF:

1. SEEK, WRITE, WRITE-CHECK
2. SEEK, READ DATA, DATA COMPARE
3. SEEK, READ HEADERS, READ 1 SECTOR W/NO HEADER COMPARE, GET STATUS
4. SEEK, READ, READ

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

- PDP-11/LSI-11 PROCESSOR WITH 16K OR MORE OF MEMORY
- CONSOLE DEVICE (LA30,LA36,VT50,ETC.)
- 1 OR 2 RL11/RLV11 CONTROLLER(S) WITH:
  1 - 8 RL01 DRIVES WITH RL01K CARTRIDGES CONTAINING A 'BAD SECTOR FILE'
  1 - 8 RL02 DRIVES WITH RL02K CARTRIDGES CONTAINING A 'BAD SECTOR FILE'
- KW11P, KW11L (OPTIONAL)
- LINE PRINTER (OPTIONAL)

1.2.2 SOFTWARE REQUIREMENTS

CZRLKA0 RL11/RLV11 RL01/RL02 PERFORMANCE EXERCISER
(FORMERLY CZRLEB0)

1.3 RELATED DOCUMENTS AND STANDARDS

RL01 USERS MANUAL (EK-RL01-UG-PRE)
XXDP USERS MANUAL

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

THE RL01/02 SUBSYSTEM SHOULD HAVE SUCCESSFULLY RUN THE FOLLOWING PROGRAMS:

CVRLAA0	RLV11 RL01 DISKLESS TEST (RLV11 ONLY)
CZRLGA0	RL11/RLV11 RL01/02 CONTROLLER TEST (PART 1)
CZRLHA0	RL11/RLV11 RL01/02 CONTROLLER TEST (PART 2)
CZRLIA0	RL01/02 DRIVE TEST (PART 1)
CZRLJA0	RL01/02 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 HOW TO RUN THIS DIAGNOSTIC

2.1.1. THE SIX STEPS OF EXECUTION

THIS DIAGNOSTIC SHOULD BE LOADED AND STARTED USING NORMAL XXDP PROCEDURES. THE START COMMAND SHOULD NOT SPECIFY AN ADDRESS, BECAUSE THE DIAGNOSTIC HAS THE PROPER TRANSFER ADDRESS CODED INTO IT.

WHEN THIS DIAGNOSTIC IS STARTED, THE FOLLOWING STEPS WILL OCCUR:

****
* STEP 1 *
****

A SHORT SERIES OF "HARDCORE QUESTIONS" WILL BE ASKED:

QUESTION	MEANING
L-CLK (L) N ?	IS THERE AN L-CLOCK?
P-CLK (L) N ?	" " " P-CLOCK?
50HZ (L) N ?	IS THE POWER 50 CYCLES (EUROPE)?
LSI (L) N ?	IS MACHINE AN LSI?
LPT (L) N ?	IS THERE A LINE PRINTER?
MEM (K) (D) 16 ?	HOW MANY K OF MEMORY ARE THERE?

THE DEFAULTS (SHOWN AFTER EACH QUESTION) CAN BE SELECTED BY HITTING CARRIAGE RETURN. IT IS POSSIBLE THAT NOT ALL OF THE QUESTIONS WILL BE ASKED: FOR EXAMPLE, IF YOU SAY "YES" TO THE L-CLOCK QUESTION, THE P-CLOCK QUESTION WILL NOT BE ASKED.

IF NEITHER P OR L CLOCK ARE ANSWERED YES THE OPERATOR WILL BE ASKED TO TYPE TWO CHARACTERS 4 SECONDS APART.

****
* STEP 2 *
****

WHEN YOU HAVE ANSWERED ALL THE HARDCORE QUESTIONS, THE DIAGNOSTIC WILL ISSUE THE PROMPT "DS-B>". FROM THIS POINT UNTIL THE TIME WHEN YOU RESTART XXDP, YOU WILL BE TALKING TO THE DIAGNOSTIC, NOT XXDP. WE WILL REFER TO THE PRESENCE OF THIS PROMPT AS BEING IN DIAGNOSTIC COMMAND MODE, AS OPPOSED TO XXDP COMMAND MODE.

AT THIS POINT YOU WILL ENTER A "START" COMMAND. THIS IS NOT THE SAME AS THE XXDP "START" COMMAND, WHICH YOU ALREADY ISSUED IN RESPONSE TO THE XXDP DOT PROMPT. THIS "START" COMMAND CAN TAKE A NUMBER OF SWITCHES AND FLAGS (ALL OPTIONAL) AND THE DETAILS OF THESE ARE SET FORTH IN "2.3 DETAILS OF COMMANDS AND SYNTAX".

HOWEVER, IN ORDER TO USE THE PROGRAM, ALL YOU NEED TO SAY IS SOMETHING LIKE THIS:

```
STA/PASS:1/FLAGS:HOE
```

THINGS TO NOTE HERE:

1. ONLY THE FIRST THREE CHARACTERS OF THIS OR ANY COMMAND AT THE "DS-B>" LEVEL NEED TO BE TYPED.

2. THE "PASS" SWITCH SPECIFIES HOW MANY PASSES YOU DESIRE. A PASS CONSISTS OF RUNNING THE FULL DIAGNOSTIC AGAINST ALL UNITS BEING TESTED (THIS WILL BE EXPLAINED SHORTLY). ONE PASS IS SPECIFIED IN THE ABOVE EXAMPLE.

3. THE "FLAGS" SWITCH MAY SPECIFY ANY OF A NUMBER OF FLAGS, BUT THE MAIN USEFUL ONES ARE:

```
LOE     LOOP ON ERROR
HOE     HALT ON ERROR
IER     INHIBIT ERROR PRINTOUT
```

THE HOE FLAG IS SPECIFIED IN THE ABOVE EXAMPLE (WE'LL SEE WHY SHORTLY).

****
* STEP 3 *
****

WHEN YOU HAVE TYPED IN A "START" COMMAND, THE DIAGNOSTIC WILL COME BACK WITH THE QUESTION "# UNITS?" TO WHICH YOU SHOULD RESPOND BY TYPING IN THE NUMBER OF DEVICES YOU WISH TO TEST.

A WORD OF WARNING HERE: THE NUMBER OF UNITS DEPENDS ON THE TARGET DEVICE OF THE DIAGNOSTIC. FOR EXAMPLE, IF THE DIAGNOSTIC IS DIRECTED AT A DISK DRIVE, THEN THE NUMBER OF UNITS WOULD BE THE NUMBER OF DRIVES TO BE TESTED. WHEREAS IF THE DIAGNOSTIC WAS DIRECTED AT THE DISK CONTROLLER, THEN THE NUMBER OF UNITS WOULD BE THE NUMBER OF CONTROLLERS. THE TARGET DEVICE OF A DIAGNOSTIC CAN ALWAYS BE DETERMINED BY INSPECTING THE "HEADER" STATEMENT NEAR THE BEGINNING OF THE SOURCE CODE. ONE OF THE OPERANDS OF THIS "HEADER" STATEMENT SHOULD BE THE DEVICE TYPE OF THE DIAGNOSTIC.

****
* STEP 4 *
****

WHEN YOU HAVE TYPED IN THE NUMBER OF UNITS TO BE TESTED, THE DIAGNOSTIC WILL ASK YOU THE "HARDWARE QUESTIONS". THE ANSWERS TO THESE QUESTIONS ARE USED TO BUILD TABLES IN CORE, CALLED "HARDWARE P-TABLES". ONE HARDWARE P-TABLE WILL BE BUILT FOR EACH UNIT TO BE TESTED.

THERE ARE SEVERAL HARDWARE QUESTIONS AND THE ENTIRE SERIES WILL BE POSED N TIMES, WHERE N IS THE NUMBER OF UNITS.

THIS REPRESENTS A NEW PHILOSOPHY IN DIAGNOSTIC ENGINEERING. DIAGNOSTICS IN THE FUTURE WILL NOT BE WRITTEN TO AUTOSIZE OR ASSUME STANDARD ADDRESSES: INSTEAD, THEY WILL ASK THE OPERATOR FOR ALL THE INFORMATION THEY NEED TO TEST THE DEVICE.

****
* STEP 5 *
****

AFTER YOU HAVE ANSWERED ALL THE HARDWARE QUESTIONS (SEC 2.5) FOR ALL THE UNITS, YOU WILL BE ASKED "CHANGE SW?" IF YOU WANT TO BE ASKED THE SOFTWARE QUESTIONS THAT DETERMINE THE BEHAVIOR OF THIS PROGRAM, TYPE "Y". IF YOU WANT TO TAKE ALL THE DEFAULTS TO THESE QUESTIONS, TYPE "N". IF YOU TYPE "Y" YOU WILL BE ASKED THE SOFTWARE QUESTIONS (SEC 2.6), AND THE ANSWERS WILL BE PUT INTO THE SOFTWARE P-TABLE IN THE PROGRAM. THE SERIES OF QUESTIONS WILL BE ASKED JUST ONCE, REGARDLESS OF THE NUMBER OF UNITS TO BE TESTED.

****
* STEP 6 *
****

AFTER YOU HAVE ANSWERED THE SOFTWARE QUESTIONS, THE DIAGNOSTIC WILL BEGIN TO EXECUTE THE HARDWARE TEST CODE. THERE ARE SEVERAL THINGS THAT CAN HAPPEN NEXT, DEPENDING ON WHETHER A HARDWARE ERROR IS ENCOUNTERED AND ALSO ON WHAT SWITCH VALUES YOU SELECTED ON THE START COMMAND. CONSIDER THE POSSIBILITIES:

1. IF NO ERROR IS ENCOUNTERED, THEN THE DIAGNOSTIC WILL SIMPLY EXECUTE THE DESIRED NUMBER OF PASSES AND RETURN TO COMMAND MODE (PROMPT DS-B>).

2. IF AN ERROR IS ENCOUNTERED, THEN ONE OF THREE THINGS HAPPENS, DEPENDING ON THE SETTINGS OF THE HOE AND LOE FLAGS.

   HOE SET: THE ERROR WILL BE REPORTED ON THE CONSOLE AND THE DIAGNOSTIC WILL RETURN TO COMMAND MODE.

   LOE SET: THE DIAGNOSTIC WILL LOOP ENDLESSLY ON THE BLOCK OF CODE THAT DETECTED THE ERROR.

   NEITHER HOE NOR LOE SET: THE ERROR WILL BE REPORTED ON THE CONSOLE AND NORMAL EXECUTION WILL RESUME AS IF NO ERROR HAD OCCURED.

2.1.2 SAMPLE RUN-THROUGH

LET'S SEE HOW ALL THIS WORKS IN A REAL SITUATION. RECALL THAT WE ENTERED THE COMMAND "STA/PASS:1/FLAGS:HOE". THIS WOULD BE A VERY TYPICAL WAY TO RUN THE DIAGNOSTIC. IF NO ERRORS ARE ENCOUNTERED, THE SINGLE REQUESTED PASS WILL BE EXECUTED AND THE PROMPT WILL BE RE-ISSUED.

IF AN ERROR IS ENCOUNTERED, THE ERROR WILL BE REPORTED AND THE PROMPT WILL BE REISSUED (BECAUSE THE HOE FLAG IS SET). AT THIS POINT THERE ARE FOUR DIFFERENT WAYS YOU CAN GET THE PROGRAM GOING AGAIN:

1. ISSUE ANOTHER "START" COMMAND (THUS GOING THRU ALL OF STEPS 2, 3, 4, 5, AND 6 AGAIN)

2. ISSUE A "RESTART" COMMAND (SAME AS START COMMAND EXCEPT THAT THE HARDWARE QUESTIONS ARE NOT ASKED)

3. ISSUE A "CONTINUE" COMMAND (EXECUTION WILL RESUME AT THE BEGINNING OF THE PARTICULAR HARDWARE TEST (MOST DIAGNOSTICS CONSIST OF A NUMBER OF THESE) THAT IT WAS IN WHEN THE ERROR HALT OCCURED. NO QUESTIONS ASKED.

4. ISSUE A "PROCEED" COMMAND: EXECUTION WILL RESUME AT THE INSTRUCTION FOLLOWING THE ERROR REPORT (THIS IS A SPECIAL COMMAND AND CAN BE ISSUED ONLY AT A HALT

THE MOST TYPICAL THING TO DO HERE IS TO ISSUE THE PROCEED, BUT WITH DIFFERENT FLAG SETTINGS. PROBABLY YOU WOULD WANT TO SAY:

```
PRO/FLAGS:IER:LOE:HOE=0
```

THIS WILL DO THE FOLLOWING:

1. TURN ON THE IER (INHIBIT ERROR PRINTOUT) FLAG
2. TURN ON THE LOE FLAG
3. TURN OFF THE HOE FLAG
4. RESUME EXECUTION AT INSTRUCTION AFTER ERROR REPORT

THE DIAGNOSTIC WILL NOW LOOP ON THE BLOCK OF CODE THAT DETECTED AND REPORTED THE ERROR, BUT NO ERROR PRINTOUT WILL OCCUR. THUS YOU CAN STUDY THE ERROR OR SCOPE IT OR WHATEVER.

WHEN YOU'VE SEEN ENOUGH, YOU MAY HIT CONTROL/C. THIS WILL TAKE YOU OUT OF THE LOOP AND PUT YOU BACK INTO COMMAND MODE. YOU NOW HAVE THREE CHOICES:

1. START
2. RESTART
3. CONTINUE

LET'S SAY YOU'VE REPAIRED THE DEFECT FOUND ABOVE AND WANT TO FINISH RUNNING THE DIAGNOSTIC. YOU WOULD TYPE

```
CON/FLAGS:HOE:IER=0:LOE=0
```

THIS WILL RESTORE THE FLAGS TO THEIR ORIGINAL VALUES AND RESUME EXECUTION AT THE BEGINNING OF THE HARDWARE TEST YOU WERE IN. IF THE ERROR DOES NOT RECUR, THE EXECUTION WILL FLOW RIGHT ON THRU TO THE NEXT ERROR OR TO END OF PASS.

IF AT END OF PASS YOU WANT TO RUN THE DIAGNOSTIC AGAIN, YOU HAVE TWO CHOICES:

1. START
2. RESTART

YOU WOULD CHOOSE ONE, DEPENDING ON WHETHER YOU WANTED TO ANSWER THE HARDWARE QUESTIONS AGAIN.

THE FULL PRINT-OUT FROM THE ABOVE DIALOGUE MIGHT LOOK LIKE THIS
(O=OPERATOR, D=DIAGNOSTIC):

```
                                        BY
                                        WHOM
                                        ENTERED:
                                        --------

.R CZRLKA                               O

CZRLK                                   D

L-CLK (L) N ? Y                         D,O
50HZ (L) N ?                            D
LSI (L) N ?                             D
LPT (L) N ?                             D
MEM (K) (D) 16 ?                        D

DS-R>STA/PASS:1/FLAGS:HOE               D,O

# UNITS (D) ? 2                        D,O

UNIT 1                                  D
RL11 (L) Y ?                            D,O
BUS ADDRESS (O) 174400 ?                D,O
VECTOR (O) 160 ?                        D,O
DRIVE (O) 0 ?                           D,O
DRIVE TYPE = RL01 (L) Y ?               D,O
BR LEVEL (O) 5 ?                        D,O

UNIT 2                                  D
RL11 (L) Y ?                            D,O
BUS ADDRESS (O) 174400 ?                D,O
VECTOR (O) 160 ?                        D,O
DRIVE (O) 0 ? 1                         D,O
DRIVE TYPE = RL01 (L) ? N               D,O (N=RL02)
BR LEVEL (O) 5 ?                        D,O

CHANGE SW (L) ? N                       D,O


CZRLK HRD ERR 00004 TST 003 SUB 002 PC:004130
ERR HLT

DS-B>PRO/FLAGS:IER:LOE:HOE=0            D,O


**********************************************************
AT THIS POINT THE DIAGNOSTIC IS LOOPING ON THE
ERROR WITHOUT PRINTING ANYTHING.  YOU CAN SCOPE
THE ERROR UNTIL YOU HAVE LOCATED IT, THEN ^C OUT
**********************************************************
```

```
^C                                          0

DS-B>CON/FLAGS:HOE:IER:LOE=0                D,0

CHANGE SW (L) ? N                           D,0

CZRLK EOP 1                                 D
^C

DS-B>RESTART/PASS:1                         D,0

CHANGE SW (L) ? N                           D,0
-------------------
-------------------
-------------------
-------------------
```

2.2 HOW TO CREATE A CHAINABLE FILE

THE DIAGNOSTIC AS RECEIVED FROM RELEASE ENGINEERING CANNOT BE RUN IN CHAIN MODE. THAT IS WHY IT BEARS THE EXTENSION "BIN" INSTEAD OF "BIC". THERE IS A WAY, HOWEVER, TO CREATE A CHAINABLE PROGRAM FROM WHAT YOU'VE GOT.

IT CONSISTS OF RUNNING THE PROGRAM WITH THE SPECIAL COMMAND "CCI" ISSUED WHERE YOU WOULD NORMALLY ISSUE A START COMMAND (TO THE PROMPT DS-B>. THIS COMMAND CAUSES THE DIAGNOSTIC TO GO THRU ALL THE QUESTIONS AND ANSWERS AND THEN TO HALT, JUST WHERE IT WOULD ORDINARILY BEGIN EXECUTION OF THE HARDWARE TEST CODE. AT THIS POINT YOU CAN DUMP THE PROGRAM AS IT SITS IN CORE TO THE LOAD MEDIUM, WITH THE NEW EXTENSION "BIC".

HERE IS A SAMPLE DIALOGUE TO ACCOMPLISH THIS:

```
.R UPD2
RESTART:  XXXXXX
*CLR
*LOAD DIAG.BIN
XFER:200   CORE:0,60602
*START 200
L-CLK (L) N ?
-------------
-------------
-------------
```

```
DS-B>CCI
  UNITS (D) ? 4
-------------
-------------
-------------
CHANGE SW (L) ? N
PTAB END:  60632

****************************************
*AT THIS POINT THE MACHINE HALTS AND*
*YOU MUST RESTART AT ADDRESS  XXXXXX*
****************************************

*HICORE 60632
CORE:  0,60632
*DUMP DK0: DIAG.BIC
```

THE RESULT OF DOING THIS IS THAT YOU CAN NOW BUILD AN XXDP CHAIN FILE CONTAINING THE XXDP COMMAND

```
    .R DIAG.BIC
```

AND THE DIAGNOSTIC WILL EXECUTE WITHOUT MANUAL INTERVENTION, USING THE ANSWERS THAT YOU GAVE IT WHEN YOU DID THE CCI COMMAND.

2.3 DETAILS OF COMMANDS AND SYNTAX

2.3.1 TABLE OF COMMAND VALIDITY

THERE ARE FOUR WAYS OF ENTERING DIAGNOSTIC COMMAND MODE, AND DIFFERENT SUBSETS OF THE DIAG COMMAND SET ARE AVAILABLE WITH EACH:

	HOW ENTERED	LEGAL COMMANDS
1.	OPERATOR ENTERED 'RUN DIAG'	START PRINT DISPLAY FLAGS ZFLAGS
2.	DIAGNOSTIC HAS FINISHED ALL ITS REQUESTED PASSES	START RESTART PRINT DISPLAY FLAGS ZFLAGS
3.	OPERATOR INTERRUPTED THE DIAGNOSTIC WITH CTRL/C	START RESTART CONTINUE PRINT DISPLAY FLAGS ZFLAGS
4.	AN ERROR WAS ENCOUNTERED WITH THE HOE FLAG SET SET	START RESTART CONTINUE PROCEED PRINT DISPLAY FLAGS ZFLAGS

2.3.2 COMMAND SYNTAX

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

THE DIAGNOSTIC IN CORE IS EXECUTED IN ACCORDANCE WITH THE SWITCHES SPECIFIED. THE MESSAGE "# UNITS?" IS PRINTED. THE START COMMAND MAY BE ISSUED WHEN DIAGNOSTIC COMMAND MODE HAS BEEN ENTERED VIA ONE OF THE FOLLOWING: A) OPERATOR TYPED "RUN DIAGNOSTIC" B) DIAGNOSTIC

FINISHED EXECUTING C) ERROR WAS ENCOUNTERED WITH HOE FLAG SET D) OPERATOR ENTERED CONTROL/C. AFTER THE OPERATOR RESPONDS TO "# UNITS?", THE HARDWARE DIALOGUE IS INITIATED. WHEN IT IS COMPLETED, THE QUESTIONS "CHANGE SW?" IS ISSUED, AND THE ANSWERS, IF GIVEN, BECOME THE NEW DEFAULTS. THEREFORE IT IS NECESSARY TO RELOAD THE PROGRAM IN ORDER TO RETURN TO THE LOAD DEFAULTS.

THE SWITCH ARGUMENTS ARE AS FOLLOWS:

"TEST-LIST" IS A SEQUENCE OF DECIMAL NUMBERS (1:2 ETC.) OR RANGES OF DECIMAL NUMBERS (1-5:8-10 ETC.) THAT SPECIFY THE TESTS TO BE EXECUTED. THE NUMBERS ARE SEPARATED BY COLONS. THE 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 THE ORDER OF SPECIFICATION. THE DEFAULT IS TO EXECUTE ALL TESTS.

"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 TEST EXECUTION. "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

IDU 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.

"EOP-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.

****
RES(TART)/TEST:TEST-LIST/PASS:PASS-CNT/FLAGS:FLAG-LIST/EOP:EOP-INCR/
UNITS:UNIT-LIST
****

THE DIAGNOSTIC IN CORE IS EXECUTED IN ACCORDANCE WITH THE SWITCHES SPECIFIED. HOWEVER, NEW "P-TABLES" ARE NOT BUILT. INSTEAD, THE ONES IN CORE ARE USED.

THE QUESTION "CHANGE SW?" IS ASKED AND THE ANSWERS GIVEN BECOME THE NEW DEFAULTS. THE COMMAND MAY BE ISSUED WHEN COMAND MODE HAS BEEN ENTERED VIA A) DIAGNOSTIC IS FINISHED B) HALT ON ERROR C) CONTROL/C.

THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:

1. "UNIT-LIST" IS A SEQUENCE OF LOGICAL UNIT NUMBERS RANGING FROM 1 THRU N (N = NUMBER OF UNITS BEING TESTED) SPECIFYING WHICH UNITS ARE TO BE TESTED. THE LOGICAL UNIT NUMBER DESIGNATES THE POSITION OF THE P-TABLE IN CORE, ACCORDING TO THE ORDER IN WHICH THEY WERE BUILT. THE UNITS SPECIFIED MUST NOT HAVE BEEN DROPPED BY THE OPERATOR DROP COMMAND. THE UNIT-LIST DEFAULTS TO "ALL THAT HAVE NOT BEEN DROPPED BY OPERATOR COMMAND". THE EFFECT OF THE UNIT-LIST LASTS UNTIL THE NEXT START (WHERE IT IS AUTOMATICALLY RESET TO "ALL") OR THE NEXT RESTART.

2. ALL UNSPECIFIED FLAG SETTINGS ARE UNCHANGED.

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

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 RE-EXECUTED. HARDWARE PARAMETERS MAY NOT BE CHANGED.

THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:

1. DEFALT FOR PASS-CNT IS THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART

2. UNSPECIFIED FLAG SETTINGS ARE UNCHANGED

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

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.

THE SWITCH ARGUMENTS ARE THE SAME AS THE START COMMAND EXCEPT:

1. UNSPECIFIED FLAG SETTINGS ARE UNCHANGED

****
CCI/TEST:TEST-LIST/PASS:PASS-CNT/FLAGS:FLAG-LIST/EOP:EOP-INCR
****

THE DIAGNOSTIC EXECUTES THRU ALL OPERATOR DIALOGUE AND HALTS AT THE HARDWARE TEST CODE. NOW THE OPERATOR CAN DUMP THE CORE IMAGE TO THE MEDIUM WITH A BIC EXTENSION.

THE BIC FILE MUST BE HANDLED DIFFERENTLY DEPENDING ON WHETHER IT IS RUN MANUALLY OR IN CHAIN MODE. IF RUN MANUALLY IT CAN BE INVOKED EITHER WITH A "START" (IN WHICH CASE IT WILL BEHAVE LIKE THE BIN FILE: THE PRE-GENERATED ANSWERS TO OPERATOR QUESTIONS WILL BE IGNORED) OR WITH A "RESTART" (IN WHICH CASE THE PRE-GENERATED OPERATOR ANSWERS WILL BE USED).

IF RUN IN CHAIN MODE, AUTOMATIC EXECUTION WILL COMMENCE IMMEDIATELY FROM THE XXDP COMMAND ".R DIAG". THE COMMAND PROMPT "DS-B>" WILL NOT BE ISSUED.

ANY SWITCHES SPECIFIED ON THE CCI COMMAND WILL CARRY OVER WHEN THE BIC FILE IS RUN IN CHAIN MODE (EXCEPT THAT UAM IS ALWAYS SET THERE) BUT WILL NOT CARRY OVER WHEN IT IS RUN MANUALLY.

TO DO A CCI ON A FULL SIZED DIAGNOSTIC (14.5K WORDS), A MACHINE SIZE LARGER THAN 16K IS REQUIRED. THE EXACT SIZE NEEDED DEPENDS ON WHICH UTILITY IS USED TO EXECUTE THE DIAGNOSTIC AT CCI TIME.

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

THE UNITS SPECIFIED ARE DROPPED FROM TESTING UNTIL THEY ARE ADDED BACK OR UNTIL A START COMMAND IS GIVEN. A DROP CANNOT BE FOLLOWED BY A PROCEED.

THERE IS ALSO A "DROP" MACRO INTERNAL TO THE DIAGNOSTIC, WHICH GIVES THE FACILITY OF AUTO-DROPPING. THE DURATION OF A PROGRAM DROP, HOWEVER, IS ONLY UNTIL THE NEXT START OR RESTART.

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

THE UNITS SPECIFIED ARE ADDED BACK (THEY MUST HAVE BEEN PREVIOUSLY DROPPED BY THE DROP COMMAND) TO THE TEST SEQUENCE. AN ADD CANNOT BE FOLLOWED BY A PROCEED.

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

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

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

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.

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

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

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

ALL FLAGS ARE CLEARED.

2.4 EXTENDED 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), 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.

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 8 RL UNITS, AND THAT THERE ARE FIVE (5) HARDWARE PARAMETERS FOR EACH (5 SLOTS IN THE P-TABLE, 5 HARDWARE QUESTIONS IN THE DIALOGUE).

FOLLOWING IS THE DIALOGUE FOR THIS 8 RLOX DRIVE SYSTEM. THIS SYSTEM HAS TWO (2) RL11 TYPE CONTROLLERS ALL TO BE SET AT "BR LEVEL" 5. THE FIRST 4 DRIVES ARE RL01'S AND THE LAST 4 DRIVES ARE RL02'S (ON THE SECOND CONTROLLER):

```
# UNITS (D) ? 8

UNIT 1
RL11 (L) Y ?
BUS ADDDRESS (O) 174400 ?
VECTOR (O) 160 ?
DRIVE (O) 0 ? 0-3
DRIVE TYPE = RL01 (L) Y ?
BR LEVEL (O) 5 ?

UNIT 5
RL11 (L) Y ?
BUS ADDRESS (O) 174400 ? 175400
VECTOR (O) 160 ? 164
DRIVE (O) 0 ? 0-3
DRIVE TYPE = RL01 (L) Y ? N
BR LEVEL (O) 5 ?
```

THE FIRST TIME THRU THE P-TABLE QUESTIONS THE DEFAULT VALUES ARE USED FOR THE CONTROLLER TYPE (QUESTION #1), CSR ADDRESS OF THE CONTROLLER (QUESTION #2), THE CONTROLLER VECTOR ASSIGNMENT (QUESTION #3), THE DRIVE TYPE (QUESTION #5), AND THE "BR LEVEL" (QUESTION #6). THE ACTUAL UNIT NUMBERS OF THE RL01'S FOR QUESTION #4 WAS ASSIGNED 0 THRU 3 FOR THE FIRST 4 P-TABLE SLOTS.

THE SECOND TIME THRU THE P-TABLE QUESTIONS (FOR THE RL02 ASSIGNMENT ON THE SECOND CONTROLLER), THE FIRST QUESTION DEFAULTED TO "RL11" TYPE CONTROLLER. THE SECOND QUESTION WAS ANSWERED TO REFLECT THE CHANGE IN CSR ADDRESS FOR THE RL02 CONTROLLER (175400). THE SECOND CONTROLLER'S VECTOR WAS ALSO CHANGED TO 164 IN QUESTION #3. THE RL02 TEST UNIT NUMBERS WERE ASSIGNED VALUES 0 TO 3 IN QUESTION #4 AND THE DRIVE TYPE WAS SET FOR RL02'S FOR THE REMAINING 4 UNITS IN QUESTION #5. THE LAST QUESTION WAS DEFAULTED USING THE "BR LEVEL" FROM THE FIRST PASS.

2.5 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) 160?

ANSWER WITH THE INTERRUPT VECTOR OF THE CONTROLLER.

DRIVE (O) 0?

ANSWER WITH THE DRIVE(S) CONNECTED TO THE CONTROLLER

DRIVE TYPE = RL01 (L) ?

ANSWER NO (N) IF DRIVE IS AN RL02

BR LEVEL (O) 5?

ANSWER WITH THE INTERRUPT PRIORITY OF THE CONTROLLER.

2.6 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.

"RETRY LMT (D) 1 ?"

THIS IS THE NUMBER OF TIMES THE PROGRAM WILL ATTEMPT A COMMAND BE-

FORE 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

"SEEK RETRY LMT (D) 1 ?"

THIS IS THE NUMBER OF RETRYS THAT WILL BE ATTEMPTED 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

"DATA DMP ON DCK ERR (L) Y ?"

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

"# OF ERR DUMPED (D) 128 ?"

THIS IS THE NUMBER OF MISCOMPARES THAT WILL BE PRINTED.

LIMITS 0 - 128

"TIME BETW REPORTS (MIN) (D) 120 ?"

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

"DROP DR ON ERR LMTS REACHED (L) Y ?"

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

"HRD ERR LMT (D) 3 ?"

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

"SFT ERR LMT (D) 10 ?"

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

"DATA MISCOMPARE LIMIT (D) 10 ?"

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

LIMITS 1 - 65,535

"SK ERR LMT (D) 3 ?"

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

LIMITS 1 - 65,535

"DR ERR LMT (D) 3 ?"

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

LIMITS 1 - 65,535

"DROP DR ON OPER LMTS REACHED (L) N ?"

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.

LIMITS Y OR N

"DATA XFER LMT (*10(10)) (D) 25000 ?"

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

LIMITS 1 - 65,535

"SK LMT (*10(3)) (D) 10000 ?"

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

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

"DO YOU WANT TO CHANGE SEEK, R/W PARAMETERS (L) N ?"

THE NORMAL OPERATION IS TO SEEK AND TRANSFER ON THE ENTIRE CARTRIDGE, CYLINDERS 0 - 255. (RL01) OR 511. (RL02), 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.

"STIPULATE R/W XFER SIZE (L) N ?"

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 NEXT TWO QUESTIONS. QUESTION IS 2.3.13.19.

LIMITS Y OR N

"MAX XFER (D) 2560 ?"

REPRESENTS THE MAXIMUM AMOUNT OF WORDS TO READ OR WRITE

LIMITS 3 - 5120

"MIN XFER (D) 3 ?"

REPRESENTS THE MINIMUM AMOUNT OF WORDS TO READ OR WRITE

LIMITS 3 - 5120

"RD ONLY (L) N ?"

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

"RAN PAT (L) Y ?"

NORMAL OPERATION SHOULD BE YES, BUT THIS PARAMETER WILL ALLOW THE WRITING OF ONLY ONE PATTERN OF EIGHT NORMAL PATTERNS. THE PATTERNS IN NEXT QUESTION.

LIMITS Y OR N

"WHICH ONE (O) 4 ?"

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

WORST CASE DATA

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

"WORDS PER SECTOR COMPARED ON READ (D) 16 ?"

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

"# OF DATA ERR RPT'D PER BUF (D) 3 ?"

THIS PARAMETER WILL LIMIT THE NUMBER OF IN CORE MISCOMPARES PRINTED. THE PROGRAM WILL CONTINUE TO COMPARE AS MANY WORDS AS SPECIFIED 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

"MAX HD (D) 1 ?"

REPRESENTS MAXIMUM HEAD TO USE IN SEEK OPERATIONS.

LIMITS 0 - 1

"MIN HD (D) 0 ?"

REPRESENTS MINIMUM HEAD TO USE IN SEEK OPERATIONS

LIMITS 0 - 1

"CHANGE VALUES OF MXCYL & MINCYL (L) Y ?"

IF NO THEN THE NEXT TWO QUESTIONS WILL BE SKIPPED

"MAX CYL (D) 511 ?"

MAXIMUM INNER CYLINDER TO BE USED IN SEEK OPERATIONS.

LIMITS 0 - 255. (RL01) OR 511. (RL02)

"MIN CYL (D) 0 ?"

MINIMUM OUTER CYLINDER TO BE USED IN SEEK OPERATIONS.

LIMITS 0 - 255. (RL01) OR 511. (RL02)

"MAX SEC (D) 0 ?"

MAXIMUM SECTOR TO START TRANSFER ON

LIMITS 0 - 39

"MIN SEC (D) 0 ?"

MINIMUM SECTOR TO START TRANSFER ON

LIMITS 0 - 39

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

"CHK DRDY (L) N ?"

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

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 BRIEF 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 POSITION,PRESENT POSITION, AND DRIVE STATUS

"VOL CHK WILL NOT RESET"

A DRIVE RESET WILL NOT RESET VOLUME CHECK BIT

"DID NOT REC'R FROM PWR UP"

DRIVE DID NOT COME BACK UP AFTER A POWER FAILURE

"DATA DMP - DATA CHECK/GARBBLED 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 THAT ARE NOT LEGAL (I.E. MORE THAN TWO CONTROLLERS)

"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

"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 TRACKING DRIFT 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 SELECTED. ANY SUBSEQUENT READ HEADER, READ OR WRITE COMMANDS WILL PRINT THIS ERROR IF THE TRACK IS NOT CORRECT. THIS ERROR WILL PRINT THE POSITION 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

"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. ONE OF THE FOLLOWING STEPS SHOULD BE ISSUED: A) STOP THE EXERCISER AND CHANGE CARTRIDGE, B) STOP THE EXERCISER AND VERIFY THE CARTRIDGE (USE THE BAD SECTOR FILE TOOL - CZRLMA) OR C) IGNORE ALL ERRORS FROM 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  DRIVE TYPE = RL0X
        *** RUNNING OR DROPPED DH:DM
        PACK SERIAL #: DDDDDDDDDD
        TOTAL SEEKS:    IIIII
        WORDS READ:     JJJJJJJJJ
        WORDS WRITTEN:  KKKKKKKKK

        ERRORS
        DRV-ER: 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
        DDDDDDDDDD - IS 10 DIGIT OCTAL SERIAL NUMBER OF PACK
        IIII    IS TOTAL NUMBER OF SEEKS SINCE START TIME 0:00:00
        JJJJ    IS TOTAL NUMBER OF WORDS READ SINCE START TIME 0:00:00
        KKKK    IS TOTAL NUMBER OF WORDS WRITTEN SINCE START TIME 0:00:00
        N       IS NUMBER OF THAT TYPE ERROR SINCE START TIME 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 15-7 - CYLINDER ADDRESS FOR TRANSFER
BIT 6 - SURFACE FOR TRANSFER
BIT 5-0 - SECTOR FOR TRANSFER (1-40.)

FOR SEEK FUNCTION

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

BIT 1 - MUST BE ZERO (0)
BIT 0 - MUST BE ONE (1)

FOR GET STATUS FUNCTION

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

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 - DRIVE TYPE IS RL02 IF SET
BIT 6 - SURFACE (0=UPPPER, 1=LOWER)
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;

3. THEN STEP 3; ELSE STEP 1

4. RANDOMLY SELECT FUNCTION FOR DRIVE
   IF SEEK/WRITE/WRITE CHECK - THEN GOTO STEP 5
   IF SEEK/READ - THEN GOTO STEP 11
   IF SEEK/READ/READ - THEN GOTO STEP 15
   IF SEEK/READ HDRS/READ W/NO HDR COMPARE/GET STATUS - THEN GOTO STEP 21

5. GET A RANDOM CYLINDER ADDRESS (NOT THE BAD SECTOR FILE)

6. SEEK TO THE SELECTED CYLINDER AND WAIT TILL COMPLETED

7. GET A RANDOM WORD COUNT FOR THE WRITE FUNCTION - MAKE SURE THAT IT WON'T OVERFLOW THE TRACK

8. GET A RANDOM DATA PATTERN TO WRITE ON THE TRACK POINTED TO

9. ISSUE THE WRITE FUNCTION AND WAIT TILL COMPLETED

10. ISSUE A WRITE CHECK FUNCTION ON THE SAME DISK ADDRESS TO COMPARE THE DATA JUST WRITTEN BY THE WRITE FUNCTION THEN GOTO STEP #1

11. GET A RANDOM CYLINDER # FOR THE SEEK

12. SEEK TO THE SELECTED CYLINDER AND WAIT TILL COMPLETED

13. GET A RANDOM WORD COUNT FOR THE READ FUNCTION - MAKE SURE IT WILL NOT OVERFLOW THE SELECTED TRACK

14. ISSUE THE READ FUNCTION AND WAIT TILL COMPLETED ...THE INTERRUPT SERVICE WILL INITIATE A DATA COMPARE ON THE DATA READ (IF THE FUNCTION IS ENABLED) THEN GOTO STEP #1

15. GET A RANDOM CYLINDER FOR THE SEEK

16. SEEK AND WAIT TILL COMPLETED

17. GET A RANDOM WORD COUNT FOR THE READ COMMAND

18. ISSUE A READ COMMAND AND WAIT TILL COMPLETED

19. GET ANOTHER RANDOM WORD COUNT FOR SAME TRACK SELECTED

20. ISSUE A SECOND READ FUNCTION AND WAIT TILL COMPLETED THEN GOTO STEP #1

21. ISSUE A SEEK TO A RANDOM CYLINDER AND WAIT TILL COMPLETED

22. ISSUE A READ HEADER FUNCTION AND WAIT TILL COMPLETED

23. ISSUE A READ DATA WITH NO HEADER COMPARE (1 SECTOR TO BE READ) AND WAIT TILL COMPLETED

24. ISSUE A GET STATUS FUNCTION THEN GOTO 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

TABLE OF CONTENTS

```
                                .TITLE  CZRLKA0 RL01/2 PERF EXER
                                .ENABLE AMA
000000                          .ENABLE ABS
                                .NLIST  ME,CND,MD

        002000                  .=2000


                                .MCALL  SVC

002000                          SVC
        000000                  SVCINS=0
        000000                  SVCTAG=0


002000                                  POINTER ALL


002000                          BGNMOD  MDHEDR
002000                                  HEADER  CZRLK,A,0,0,0,0,RL01,1
002000      103                         .ASCII /C/
002001      132                         .ASCII /Z/
002002      122                         .ASCII /R/
002003      114                         .ASCII /L/
002004      113                         .ASCII /K/
002005      000                         .BYTE   0
002006      000                         .BYTE   0
002007      000                         .BYTE   0
002010      101                         .ASCII  /A/
002011      060                         .ASCII  /0/
002012  000000                          .WORD   0
002014  000000                          .WORD   0
002016  030000                          .WORD   L$HARD
002020  030154                          .WORD   L$SOFT
002022  010230                          .WORD   L$HW
002024  010246                          .WORD   L$SW
002026  032114                          .WORD   L$LAST
002030  000000                          .WORD   0
002032  000000                          .WORD   0
002034  000001                          .WORD   1
002036  000000                          .WORD   0
002040  010350                          .WORD   L$DISPATCH
002042  000000                          .WORD   0
002044  000000                          .WORD   0
002046  000000                          .WORD   0
002050      002                         .BYTE   C$REVISION
002051      002                         .BYTE   C$EDIT
002052  000000                          .WORD   0
002054  000000                          .WORD   0
002056  000000                          .WORD   0
002060  000000                          .WORD   0
002062  000000                          .WORD   0
002064  002114                          .WORD   L$DVTYP
002066  010352                          .WORD   L$RPT
002070  002112                          .WORD   L$DR
002072  002112                          .WORD   L$DRST
002074  012060                          .WORD   L$AU
```

```
002076  012144                          .WORD   L$DU
002100  000014                          .WORD   14
002102  000000                          .WORD   0
002104  010430                          .WORD   L$INIT
002106  011714                          .WORD   L$CLEAN
21
22 002110                               ENDMOD
23
24 002110                               DEVREG
002110  000000                          .WORD   0
                                        .BLKW
25 002114                               DEVTYP  <RL01,RL02>
002114  122     114     060             .ASCIZ  /RL01,RL02/
002117  061     054     122
002122  114     060     062
002125  000
                                        .EVEN
26
27                              .SBTTL  BIT AND OFFSET DEFINITIONS
28
29                              ;DEFINITIONS
30
31
32 002126                       BGNMOD  GLBEQAT
33
34 002126                       EQUALS
35
36      000000                  CS=0                    ;CONTROL AND STATUS OFFSET
37      000002                  BA=2                    ;BUSADDRESS OFFSET
38      000004                  DA=4                    ;DISK ADDRESS OFFSET
39      000006                  MP=6                    ;MULTI PURPOSE OFFSET
40                              ;CONSTANT OFFSETS FOR INDIVDUAL DRIVE BUFFERS
41                              ;THE ONLY POSITION THAT IS CRITICAL IS THAT OF
42                              ;'"'PRPOS"'  I I M U S T (MUST) BE THE LAST ENTRY OF THE BUFFER
43
44      000000                  SKCNT=0                 ;SEEK OPERATION COUNT
45      000002                  RXFR1=2                 ;READ OPERATION COUNT (BITS) LOW ORDER
46      000004                  RXFR2=4                 ;  "       "       "     "    HIGH ORDER
47      000006                  WXFR1=6                 ;WRITE OPERATION COUNT (BITS) LOW ORDER
48      000010                  WXFR2=10                ;  "       "       "     "    HIGH ORDER
49      000012                  ERRCNT=12               ;ERROR COUNT - HARD
50      000014                  SFTCNT=14               ;ERROR COUNT - SOFT
51      000016                  SKECNT=16               ;SEEK ERROR COUNT
52      000020                  DERCNT=20               ;DRIVE ERROR COUNT
53      000022                  DCRCER=22               ;DATA CRC ERROR COUNT
54      000024                  HCRCER=24               ;HEADER CRC ERROR COUNT
55      000026                  DLTCNT=26               ;DATA LATE ERROR COUNT
56      000030                  OPICNT=30               ;OPERATION INCOMPLETE ERROR COUNT
57      000032                  HNFERR=32               ;HEADER NOT FOUND ERROR COUNT
58      000034                  NXMCNT=34               ;NON EXISTANT MEMORY ERROR COUNT
59      000036                  RETRY=36                ;PRESENT RETRY NUMBER
60      000040                  BDA=40                  ;  "     DISK ADDRESS CONTENTS
61      000042                  BMP=42                  ;PRESENT MULTIPURPOSE CONTENTS
62      000044                  FUNC=44                 ;LAST FUNCTION LOADED
63      000046                  BCSADR=46               ;CSR IMAGE OF LAST COMMAND
64      000050                  LSTHDR=50               ;LAST POSITION ON DISK
65      000052                  RTYPE=52                ;ERROR ON WHICH RECOVERY IS BEING TRIED
```

```
                        SKCNT1=54               ;LOW SEEK COUNT
                        PRFLGS=56               ;INTERNAL FLAGS
                        RXFR3=60                ;THIRD ORDER READ COUNT
                        WXFR3=62                ;THIRD ORDER WRITE COUNT
                        LSTDA=64                ;DISK ADDRESS AT SOFT ERROR
                        DIFWD=66                ;LAST DIFFERENCE WORD OF SEEK
                        DPHOUR=70               ;HOUR OF DRIVE DROPPED
                        DPMIN=71                ;MINUTE OF DRIVE DROPPED
                        TRERR=72                ;TRACKING ERRORS COUNT
                        DATCER=74               ;DATA CMP ERRORS
                        DOWCK=76                ;PERFORM WRITE CHECK
                        SERNM1=100              ;SERIAL NUMBER OF CARTRIDGE
                        SERNM2=102              ;SERIAL NUMBER OF CARTRIDGE
                        DCS=104                 ;CSR ADDRESS
                        DRSEL=106               ;DRIVE SELECT BITS(8,9,10)
                        BBA=110                 ;PRESENT BUS ADDRESS CONTENTS
                        BSECPT=112              ;POINTER TO BAD SECTOR FILE
                        RSEEK=114               ;SEEK IN PROCESS OF RECOVERY
                        SOFTCS=116              ;CSR OF SOFT ERROR
                        TDR=120
                        WRIPG=122               ;WRITE IN PROGRESS FLAG
                        PRPOS=124               ;PRESENT POSITION ON DISK

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

```
                        ;OFFSET FOR HARDWARE P-TABLE

000000                  CSR=0
000002                  VECT=2
000004                  PRIOR=4
000006                  TYPDR=6
000010                  DRBT=10
000012                  CNT=12

                        ;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
000076                  ANS=76


002126                          ENDMOD

                        ;

                        .SBTTL  GLOBAL DATA AND CONSTANTS

002126                  BGNMOD  GLBDAT

002126  000000          RECNT:  .WORD   0               ;READ ERROR COUNT
002130  000000          RWCNT:  .WORD   0               ;R/W ERROR COUNT
002132  000000          WHY:    .WORD   0               ;REASON FOR DROPPING DRIVE
```

```
180 002134  000000          TSTDRV: .WORD   0               ;COPY OF SELECTED DRIVE FOR TESTING
181 002136  000             DRUT:   .BYTE   0               ;DRIVES UNDER TEST
182 002137  000             DRPRS:  .BYTE   0               ;DRIVES PRESENT
183 002140  000000          T.DRIVE: .WORD  0               ;TYPE OF DRIVE FROM P-TABLE
184 002142  000000          SYSMSK: .WORD   0               ;MASK FOR 0-7 DRIVES
185 002144  176543          HINUM:  .WORD   176543          ;PRIME FOR RANDOM
186 002146  123456          LONUM:  .WORD   123456          ;NUMBER GENERATOR
187 002150  100177          CYLMSK: .WORD   100177          ;MASK FOR CYLINDER ONLY
188 002152  100077          SECMSK: .WORD   100077          ;MASK OUT SECTOR BITS
189 002154  000177          CMSK:   .WORD   000177
190 002156  000077          SMSK:   .WORD   000077
191 002160  000000          WRINIT: .WORD   0               ;WRITE INIT FLAG
192 002162  000000          WRPOS:  .WORD   0               ;WRITE UNIT FLAG
193 002164  000000          CYL:    .WORD   0               ;CYLINDER #
194 002166  000000          SUR:    .WORD   0               ;SURFACE #
195 002170  000000          SEC:    .WORD   0               ;SECTOR #
196 002172  000000          REGEN:  .WORD   0               ;REGEN FLAG FOR BUFFERS
197 002174  000000          KILLDC: .WORD   0               ;INHIBIT DATA COMP FLAG
198                          ;
199                          ;THE FOLLOWING LOCATIONS ARE CLEARED AS A GROUP (DOWN TO 'STFLG')
200                          ;THEREFORE DON'T INSERT ANY CONSTANTS
201                          ;
202 002176  174400          CNTLR1: .WORD   174400          ;CSR OF CONTROLLER 1 (LUN 0-3)
203 002200  000000          CNTLR2: .WORD   0               ;CSR OF CONTROLLER 2 (LUN 4-7)
204 002202  000000          LSTDR1: .WORD   0               ;BUFFER POINTER OF DRIVE
205 002204  000000          LSTDR2: .WORD   0               ;BUFFER POINTER OF DRIVE
206 002206  000000          BCSR:   .WORD   0               ;CSR FROM P-TABLE
207 002210  000000          BVEC:   .WORD   0               ;VECTOR "    "
208 002212  000000          BPRIOR: .WORD   0
209 002214  000000          BDRSEL: .WORD   0               ;DRIVE  "    "
210 002216  000000          HDRFND: .WORD   0               ;FLAG TO INDICATE HDR IN BAD LIST
211 002220  000000          CHKSEC: .WORD   0               ;SECTOR OF ERROR - USED BY BAD SECTOR LOCATION
212 002222  000000          DECNT:  .WORD   0               ;DATA ERROR COUNT
213 002224  000000          TEMP0:  .WORD   0               ;TEMP LOCATION
214 002226  000000          TEMP1:  .WORD   0               ;TEMP LOCATION
215 002230  000000          TEMP2:  .WORD   0               ;TEMP LOCATION
216 002232  000000          TEMP3:  .WORD   0               ; "    "
217 002234  000000          TEMP4:  .WORD   0               ; "    "
218 002236  000000          TEMP5:  .WORD   0               ; "    "
219 002240  000000          TEMP6:  .WORD   0               ; "    "
220 002242  000000          TEMP7:  .WORD   0               ; "    "
221 002244  000000          TEMP8:  .WORD   0               ; "    "
222 002246  000000          TEMP9:  .WORD   0               ; "    "
223 002250  000160          VECT1:  .WORD   160             ;VECTOR OF FIRST CONTROLLER
224 002252  000000          VECT2:  .WORD   0               ;VECTOR  "  2ND
225 002254  000000          PRIOR1: .WORD   0
226 002256  000000          PRIOR2: .WORD   0
227 002260  000000          GDDAT:  .WORD   0               ;
228 002262  000000          RNTEMP: .WORD   0               ;    "    "
229 002264  000000          INTERVAL:       .WORD   0       ;TIME BETWEEN REPORTS
230 002266  000000          LSTTIM: .WORD   0               ;LAST TIME ON SYSTEM CLOCK
231 002270  000000          SECOND: .WORD   0               ;SECONDS OF SYSTEM CLOCK
232 002272  000000          MINUTE: .WORD   0               ;MINUTES OF SYSTEM CLODK
233 002274  000000          HOUR:   .WORD   0               ;HOURS OF SYSTEM CLOCK
234 002276  000000          E.CS:   .WORD   0               ;IMAGES OF REGISTERS
235 002300  000000          E.BA:   .WORD   0               ;ON INTERRUPT
236 002302  000000          E.DA:   .WORD   0               ;
```

```
                                E.MP:   .WORD   0               ;
                                E.MP1:  .WORD   0               ;
                                E.MP2:  .WORD   0               ;
                                C.HDR:  .WORD   0               ;CURRENT HEADER - FOR ERROR REPORT
                                SYSCLK: .WORD   0               ;FLAG INDICATING PRESENSE OF SYSTEM CLOCK
                                BUF1:   .WORD   0               ;BUFFER FOR  FIRST CONTROLLER
                                BUF2:   .WORD   0               ;BUFFER FOR SECOND CONTROLLER
                                MAXWC:  .WORD   0               ;MAX WORD COUNT DETERMINED BY CORE
                                UUT:    .WORD   0               ;NUMBER OF UNITS ON SYSTEM
                                PWRFLG: .WORD   0               ;POWER FAIL INDICATOR
                                TRPFLG: .WORD   0               ;TRAP OCCURANCE FLAG
                                STFLG:  .WORD   0               ;START FLAG
                                ;
                                ;END OF MASS CLEAR
                                ;
                                CNIFLG: .WORD   0               ;CONTINUE FLAG
                                FASCII: .WORD   0               ;ASCII MESSAGE OF FUNCTION
                                FASPNT: .WORD   0               ;POINTER
                                DWCNT:  .WORD   0               ;ERROR COUNT
                                DWCNT1: .WORD   0               ;ERROR COUNT
                                ERRVEC: .WORD   4               ;ERROR VECTOR
                                ST1:    .WORD   34              ;STATES ALLOWED
                                ST2:    .WORD   35              ;STATES ALLOWED
                                OPCALL: .WORD   0
                                INCALL: .WORD   0

                                        ENDMOD


                                .SBTTL  GLOBAL MESSAGES

                                BGNMOD  GLBTXT

                                ;GLOBAL TEXT


                                TIME:   .ASCIZ  "TIME: "
                                MRLCS:  .ASCIZ  " RLCS: "
                                CRLCS:  .ASCIZ  " (RLCS): "
                                MFUNC:  .ASCIZ  ">> FUNCTION: "
                                CRLBA:  .ASCIZ  " (RLBA): "
                                CRLDA:  .ASCIZ  " (RLDA): "
                                CRLMP:  .ASCIZ  " (RLMP): "

                                DIFMSG: .ASCIZ  /DIF WD: /
                                CART:   .ASCIZ  /PACK SERIAL #: /
                                NOCRDY: .ASCIZ  /NO CRDY/
                                DNRDY:  .ASCIZ  /DR NOT RDY/
                                NORDY:  .ASCIZ  %DR NOT RDY W/O DR ERR%
                                PRGER:  .ASCIZ  /BUG/
                                NWRTS:  .ASCIZ  /INIT WR OF SEC BAD/
                                SMSG:   .ASCIZ  / SECTOR: /
                                EXHAUS: .ASCIZ  /NO GOOD HDR/
                                UDERR:  .ASCIZ  /UNDIAGNOSABLE ERR/
                                MSKER:  .ASCIZ  /SEEK ERR/
                                MSFER:  .ASCIZ  /SOFT ERR ENC'D/
```

```
    297 002716  104  122  040  DRVER:  .ASCIZ  /DR ERR/
    298 002725  104  122  040  MDERS:  .ASCIZ  /DR ERR WILL NOT RESET/
    299 002753  104  122  040  MDSER:  .ASCIZ  /DR STAT ERR/
    300 002767  126  117  114  MVCER:  .ASCIZ  /VOL CHK WILL NOT CLR/
    301 003014  127  122  040  WGEST:  .ASCIZ  /WR GATE ERR WILL NOT RESET/
    302 003047  104  122  040  MRDER:  .ASCIZ  /DR ERR - RECOVERED/
    303 003072  104  101  124  MDCER:  .ASCIZ  /DATA CMP ERR/
    304 003107  110  101  122  MHDER:  .ASCIZ  /HARD ERROR/
    305 003122  104  101  124  DMPDCK: .ASCIZ  /DATA DUMP - DCK/
    306 003142  124  122  101  TRACK:  .ASCIZ  /TRACKING ERR/
    307 003157  110  122  104  ERLMTM: .ASCIZ  /HRD ERR LMT EXC'D/
    308 003201  123  113  040  SERLMT: .ASCIZ  /SK ERR LMT EXC'D/
    309 003222  123  106  124  SFEMSG: .ASCIZ  /SFT ERR LMT EXC'D/
    310 003244  104  101  124  DCDMSG: .ASCIZ  /DATA ERR LMT EXC'D/
    311 003267  104  122  040  DERMSG: .ASCIZ  /DR ERR LMT EXC'D/
    312 003310  102  125  106  OVER:   .ASCIZ  /BUFFER CHOSEN TOO BIG - WAS /
    313 003345  122  105  121  REQ:    .ASCIZ  /REQ BY OPR/
    314 003360  105  130  110  SEXHAU: .ASCIZ  /EXH'D RETRY ON SEEK/
    315 003404  110  104  123  UNLOAD: .ASCIZ  /HDS NOT UNLD ON ERR/
    316 003430  104  122  040  NOLOAD: .ASCIZ  /DR WLD NOT LD/
    317 003446  117  120  105  SOPLMT: .ASCIZ  /OPER LMTS EXC'D/
    318 003466  107  101  122  NOREV:  .ASCIZ  /GARBBLED DATA - CAN'T CHECK IT/
    319 003525  115  117  122  MBDMSC: .ASCIZ  /MORE THAN 16 BAD SECTORS/
    320 003556  116  117  040  HWSEC:  .ASCIZ  /NO FACTORY FILE/
    321 003576  116  117  040  SWSEC:  .ASCIZ  /NO FIELD FILE/
    322 003614  120  055  124  MPT:    .ASCIZ  /P-TABLE: /
    323 003626  111  114  114  ILLEG:  .ASCIZ  /ILL P-TABLE/
    324 003642  040  126  105  MVEC:   .ASCIZ  / VECTOR: /
    325 003654  116  117  040  NODRIV: .ASCIZ  /NO DRIVES/
    326 003666  040  104  122  DRNM:   .ASCIZ  / DRIVE: /
    327 003677  040  114  123  LPS:    .ASCIZ  / LST POS: /
    328 003712  105  130  120  EPS:    .ASCIZ  /EXP POS: /
    329 003724  040  122  105  RPS:    .ASCIZ  / REC POS: /
    330 003737  104  122  040  NOPWR:  .ASCIZ  /DR DID REC'R FROM PWR UF/
    331 003770  101  124  040  BUSAD:  .ASCIZ  /AT BUS ADDR: /
    332 004006  122  105  124  MRT:    .ASCIZ /RETRYS: /
    333 004017  040  105  122  ERT:    .ASCIZ  / ERROR TYPE: /
    334 004035  123  124  101  MST:    .ASCIZ  /STATUS WAS: /
    335 004052  040  123  110  MST1:   .ASCIZ  / SHOULD BE: /
    336 004067  040  122  105  RT1:    .ASCIZ  / RETRIES ATTEMPTED/
    337 004112  040  105  130  EXP:    .ASCIZ  / EXP'D: /
    338 004123  040  122  105  RCD:    .ASCIZ  / REC'D: /
    339 004134  104  122  111  DROP:   .ASCIZ  /DRIVE DROPPED/
    340 004152  040  110  116  MTHNF:  .ASCIZ  / HNF/
    341 004157  040  110  103  MTHCRC: .ASCIZ  / HCRC/
    342 004165  040  104  103  MTDCRC: .ASCIZ  / DCK/
    343 004172  040  104  114  MTDLT:  .ASCIZ  / DLT/
    344 004177  040  117  120  MTOPI:  .ASCIZ  / OPI/
    345 004204  040  116  130  MTNXM:  .ASCIZ  / NXM/
    346 004211  040  104  122  MTDRV:  .ASCIZ  / DRV/
    347 004216  124  105  123  MSTART: .ASCIZ  /TESTING STARTED/
    348 004236  127  122  111  MSWRPK: .ASCIZ  /WRITING PACK /
    349 004254  120  101  103  NORDDC: .ASCIZ  /PACK NOT FULLY INIT'D...DATA COMPARE INHIBITTED/
    350 004334  103  125  122  ERRHDR: .ASCIZ  /CURRENT POSITION (HDR) = /
    351                         ;
    352                         ;THIS LIST OF ASCII TEXT IS USED AS A TABLE FOR PRINTING
    353                         ;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
                                        ;FILL BYTE (TOTAL 8(EIGHT) BYTES) LONG.  USED IN LINE1
                                        ;SUBROUTINE.....................................
                                        ;.....................................................

004366      040     127     122  MTCR:   .ASCIZ  / WRCHK /
004376      040     107     124  MTGS:   .ASCIZ  / GTSTAT/
004406      040     123     105  MTSK:   .ASCIZ  / SEEK  /
004416      040     122     104  MTRH:   .ASCIZ  / RDHDR /
004426      040     127     122  MTWR:   .ASCIZ  / WRITE /
004436      040     122     105  MTRD:   .ASCIZ  / READ  /
004446      040     122     104  MTRNH:  .ASCIZ  / RD-NHD/

                                        ;.....................................................
                                        ;END OF LIST NOW YOU CAN PUT ANY THING YOU WANT HERE
                                        ;.....................................................




                                        .EVEN

004456                                          ENDMOD

                                        .SBTTL  ERROR MESSAGES

004456                                  BGNMOD  GLBERR

                                                                        ;GENERAL ERROR REPORT

004456                                  BGNMSG  ERR1
004456  004737  005716                        JSR     PC,LINE3
004462                                        ENDMSG
004462                                  L10000:
004462  104023                                EMT     C$MSG

                                                                        ;MIS-SEEK ERROR REPORT

004464                                  BGNMSG  ERR2
004464  004737  005716                        JSR     PC,LINE3
004470                                        PRINTB  #FMT4,#DIFMSG,DIFWD(R4),#LPS,LSTHDR(R4),#EPS,PRPOS(R4),#RPS,R1
004470  010146                                MOV     R1,-(SP)
004472  012746  003724                        MOV     #RPS,-(SP)
004476  016446  000124                        MOV     PRPOS(R4),-(SP)
004502  012746  003712                        MOV     #EPS,-(SP)
004506  016446  000050                        MOV     LSTHDR(R4),-(SP)
004512  012746  003677                        MOV     #LPS,-(SP)
004516  016446  000066                        MOV     DIFWD(R4),-(SP)
004522  012746  002465                        MOV     #DIFMSG,-(SP)
004526  012746  006472                        MOV     #FMT4,-(SP)
004532  012746  000011                        MOV     #11,-(SP)
004536  010600                                MOV     SP,R0
004540  104014                                EMT     C$PNTB
004542  062706  000024                        ADD     #24,SP
004546                                        ENDMSG
004546                                  L10001:
```

```
     004546  104023                          EMT     C$MSG
398
399                                                          ;SOFT.ERROR RECOVERABLE ERROR REPORT
400 004550                          BGNMSG  ERR3
401 004550  004737  005402                  JSR     PC,LINE1
402 004554                                  PRINTB  #FMT2A,#CRLCS,SOFTCS(R4),#CRLBA,@BBA(R4),#CRLDA,LSTDA(R4)
    004554  016446  000064                  MOV     LSTDA(R4),-(SP)
    004560  012746  002441                  MOV     #CRLDA,-(SP)
    004564  017446  000110                  MOV     @BBA(R4),-(SP)
    004570  012746  002427                  MOV     #CRLBA,-(SP)
    004574  016446  000116                  MOV     SOFTCS(R4),-(SP)
    004600  012746  002377                  MOV     #CRLCS,-(SP)
    004604  012746  006323                  MOV     #FMT2A,-(SP)
    004610  012746  000007                  MOV     #7,-(SP)
    004614  010600                          MOV     SP,R0
    004616  104014                          EMT     C$PNTB
    004620  062706  000020                  ADD     #20,SP
403 004624  016437  000064  002224          MOV     LSTDA(R4),TEMP0 ;GET THE ADDRESS TO PRINT
404 004632  004537  006076                  JSR     R5,TELCYL       ;CONVERT FOR PRINTING
405 004636                                  PRINTB  #FMT5,#MRT,RETRY(R4),#ERT,RTYPE(R4)
    004636  016446  000052                  MOV     RTYPE(R4),-(SP)
    004642  012746  004017                  MOV     #ERT,-(SP)
    004646  016446  000036                  MOV     RETRY(R4),-(SP)
    004652  012746  004006                  MOV     #MRT,-(SP)
    004656  012746  006525                  MOV     #FMT5,-(SP)
    004662  012746  000005                  MOV     #5,-(SP)
    004666  010600                          MOV     SP,R0
    004670  104014                          EMT     C$PNTB
    004672  062706  000014                  ADD     #14,SP
406 004676                                  ENDMSG
    004676                          L10002:
    004676  104023                          EMT     C$MSG
407
408                                                          ;GET STATUS ERROR REPORT
409
410 004700                          BGNMSG  ERR4
411 004700  004737  005716                  JSR     PC,LINE3
412 004704                                  PRINTB  #FMT6,#MST,E.MP,#MST1,ST1,ST2
    004704  013746  002352                  MOV     ST2,-(SP)
    004710  013746  002350                  MOV     ST1,-(SP)
    004714  012746  004052                  MOV     #MST1,-(SP)
    004720  013746  002304                  MOV     E.MP,-(SP)
    004724  012746  004035                  MOV     #MST,-(SP)
    004730  012746  006541                  MOV     #FMT6,-(SP)
    004734  012746  000006                  MOV     #6,-(SP)
    004740  010600                          MOV     SP,R0
    004742  104014                          EMT     C$PNTB
    004744  062706  000016                  ADD     #16,SP
413 004750                                  ENDMSG
    004750                          L10003:
    004750  104023                          EMT     C$MSG
414
415
416                                                          ;DATA ERROR SUMMARY
417
418 004752                          BGNMSG  ERR6
419 004752  004737  005606                  JSR     PC,LINE2
```

```
420 004756  016400  000042                  MOV     BMP(R4),R0
421 004762                                  PRINTB  #FMT9A,DECNT,R0
    004762  010046                          MOV     R0,-(SP)
    004764  013746  002222                  MOV     DECNT,-(SP)
    004770  012746  006651                  MOV     #FMT9A,-(SP)
    004774  012746  000003                  MOV     #3,-(SP)
    005000  010600                          MOV     SP,R0
    005002  104014                          EMT     C$PNTB
    005004  062706  000010                  ADD     #10,SP
422 005010                                  ENDMSG
    005010                          L10004:
    005010  104023                          EMT     C$MSG
423
424                                                                 ;NON RECOVERABLE ERROR REPORT
425
426 005012                          BGNMSG  ERR7
427 005012                                  PRINTB  #FMT8,RETRY(R4),#RT1
    005012  012746  004067                  MOV     #RT1,-(SP)
    005016  016446  000036                  MOV     RETRY(R4),-(SP)
    005022  012746  006603                  MOV     #FMT8,-(SP)
    005026  012746  000003                  MOV     #3,-(SP)
    005032  010600                          MOV     SP,R0
    005034  104014                          EMT     C$PNTB
    005036  062706  000010                  ADD     #10,SP
428 005042  004737  005716                  JSR     PC,LINE3
429 005046                                  ENDMSG
    005046                          L10005:
    005046  104023                          EMT     C$MSG
430
431                                                                 ;BAD DATA COMPARE ERROR REPORT
432
433 005050                          BGNMSG  ERR8
434 005050  004737  005606                  JSR     PC,LINE2
435 005054  016437  000040  002224          MOV     BDA(R4),TEMPO
436 005062  004537  006076                  JSR     R5,TELCYL                ;REPORT THE CYL # & SECTOR/HEAD
437 005066                                  PRINTB  #FMT10A,#CRLBA,@BBA(R4),#CRLDA,BDA(R4),#EXP,GDDAT,#RCD,(R2)
    005066  011246                          MOV     (R2),-(SP)
    005070  012746  004123                  MOV     #RCD,-(SP)
    005074  013746  002260                  MOV     GDDAT,-(SP)
    005100  012746  004112                  MOV     #EXP,-(SP)
    005104  016446  000040                  MOV     BDA(R4),-(SP)
    005110  012746  002441                  MOV     #CRLDA,-(SP)
    005114  017446  000110                  MOV     @BBA(R4),-(SP)
    005120  012746  002427                  MOV     #CRLBA,-(SP)
    005124  012746  006761                  MOV     #FMT10A,-(SP)
    005130  012746  000011                  MOV     #11,-(SP)
    005134  010600                          MOV     SP,R0
    005136  104014                          EMT     C$PNTB
    005140  062706  000024                  ADD     #24,SP
438 005144                                  PRINTB  #FMT10B,R2
    005144  010246                          MOV     R2,-(SP)
    005146  012746  007032                  MOV     #FMT10B,-(SP)
    005152  012746  000002                  MOV     #2,-(SP)
    005156  010600                          MOV     SP,R0
    005160  104014                          EMT     C$PNTB
    005162  062706  000006                  ADD     #6,SP
439 005166                                  ENDMSG
```

```
                005166
                005166  104023                  L10006:
                                                        EMT     C$MSG
440                                                                             ;DRIVE ERROR
441
442             005170                          BGNMSG  ERR9
443
444             005170  004737  005716                  JSR     PC,LINE3
445             005174                                  PRINTB  #FMT13,#MST,R1,#LPS,LSTHDR(R4)
                005174  016446  000050                  MOV     LSTHDR(R4),-(SP)
                005200  012746  003677                  MOV     #LPS,-(SP)
                005204  010146                          MOV     R1,-(SP)
                005206  012746  004035                  MOV     #MST,-(SP)
                005212  012746  007070                  MOV     #FMT13,-(SP)
                005216  012746  000005                  MOV     #5,-(SP)
                005222  010600                          MOV     SP,R0
                005224  104014                          EMT     C$PNTB
                005226  062706  000014                  ADD     #14,SP
446             005232                                  ENDMSG
                005232                          L10007:
                005232  104023                          EMT     C$MSG
447
448
449                                                                             ;INVALID ENTRY IN P-TABLE REPORT
450
451             005234                          BGNMSG  ERR10
452             005234                                  PRINTB  #FMT11,#MPT,R1,#MRLCS,BCSR,#MVEC,BVEC
                005234  013746  002210                  MOV     BVEC,-(SP)
                005240  012746  003642                  MOV     #MVEC,-(SP)
                005244  013746  002206                  MOV     BCSR,-(SP)
                005250  012746  002367                  MOV     #MRLCS,-(SP)
                005254  010146                          MOV     R1,-(SP)
                005256  012746  003614                  MOV     #MPT,-(SP)
                005262  012746  007040                  MOV     #FMT11,-(SP)
                005266  012746  000007                  MOV     #7,-(SP)
                005272  010600                          MOV     SP,R0
                005274  104014                          EMT     C$PNTB
                005276  062706  000020                  ADD     #20,SP
453             005302                                  ENDMSG
                005302                          L10010:
                005302  104023                          EMT     C$MSG
454
455
456             005304                          BGNMSG  ERR12
457
458             005304  004737  005716                  JSR     PC,LINE3
459
460             005310                                  ENDMSG
                005310                          L10011:
                005310  104023                          EMT     C$MSG
461
462             005312                          BGNMSG  ERR13
463             005312  004737  005716                  JSR     PC,LINE3
464             005316  016403  000104                  MOV     DCS(R4),R3
465             005322  016337  000006  002304          MOV     MP(R3),E.MP     ;GET HEADER
466             005330                                  PRINTB  #FMT14C         ;CRLF
                005330  012746  007150                  MOV     #FMT14C,-(SP)
                005334  012746  000001                  MOV     #1,-(SP)
```

```
        005340  010600                          MOV     SP,R0
        005342  104014                          EMT     C$PNTB
        005344  062706  000004                  ADD     #4,SP
    467 005350                                  PRINTB  #FMT12,#ERRHDR,C.HDR ;PRINT THE HEADER MESSAGE
        005350  013746  002312                  MOV     C.HDR,-(SP)
        005354  012746  004334                  MOV     #ERRHDR,-(SP)
        005360  012746  007060                  MOV     #FMT12,-(SP)
        005364  012746  000003                  MOV     #3,-(SP)
        005370  010600                          MOV     SP,R0
        005372  104014                          EMT     C$PNTB
        005374  062706  000010                  ADD     #10,SP
    468 005400                                  ENDMSG
        005400                          L10012:
        005400  104023                          EMT     C$MSG
    469
    470 005402  016437  000044  002340  LINE1:  MOV     FUNC(R4),FASPNT         ;GET FUNCTION
    471 005410  012737  004366  002336          MOV     #MTCR,FASCII            ;FIRST FUNCTION ASCIZ
    472 005416  042737  000100  002340          BIC     #INTEN,FASPNT           ;CLEAR INTERRUPT ENABLE
    473 005424  006237  002340                  ASR     FASPNT                  ;ALIGN - NOW = 1 TO 7
    474 005430  005337  002340          1$:     DEC     FASPNT                  ;DOWN COUNT FUNCTION
    475 005434  001404                          BEQ     2$                      ;FOUND?
    476 005436  062737  000010  002336          ADD     #8.,FASCII              ;NO NEXT ONE
    477 005444  000771                          BR      1$                      ;LOOP
    478
    479 005446                          2$:     PRINTB  #FMT10,#TIME,HOUR,MINUTE,SECOND,#MRLCS,DCS(R4),#DRNM,<B,DRSEL+1(R4)>
        005446  005046                          CLR     -(SP)
        005450  156416  000107                  BISB    DRSEL+1(R4),(SP)
        005454  012746  003666                  MOV     #DRNM,-(SP)
        005460  016446  000104                  MOV     DCS(R4),-(SP)
        005464  012746  002367                  MOV     #MRLCS,-(SP)
        005470  013746  002270                  MOV     SECOND,-(SP)
        005474  013746  002272                  MOV     MINUTE,-(SP)
        005500  013746  002274                  MOV     HOUR,-(SP)
        005504  012746  002360                  MOV     #TIME,-(SP)
        005510  012746  006725                  MOV     #FMT10,-(SP)
        005514  012746  000011                  MOV     #11,-(SP)
        005520  010600                          MOV     SP,R0
        005522  104014                          EMT     C$PNTB
        005524  062706  000024                  ADD     #24,SP
    480 005530                                  PRINTB  #FMTDT,TDR(R4)
        005530  016446  000120                  MOV     TDR(R4),-(SP)
        005534  012746  007552                  MOV     #FMTDT,-(SP)
        005540  012746  000002                  MOV     #2,-(SP)
        005544  010600                          MOV     SP,R0
        005546  104014                          EMT     C$PNTB
        005550  062706  000006                  ADD     #6,SP
    481 005554                                  PRINTB  #FMT1A,#MFUNC,FASCII
        005554  013746  002336                  MOV     FASCII,-(SP)
        005560  012746  002411                  MOV     #MFUNC,-(SP)
        005564  012746  006273                  MOV     #FMT1A,-(SP)
        005570  012746  000003                  MOV     #3,-(SP)
        005574  010600                          MOV     SP,R0
        005576  104014                          EMT     C$PNTB
        005600  062706  000010                  ADD     #10,SP
    482 005604  000207                          RTS     PC
    483
    484 005606                          LINE2:  PRINTB  #FMT10,#TIME,HOUR,MINUTE,SECOND,#MRLCS,DCS(R4),#DRNM,<B,DRSEL+1(R4)>
```

```
        005606  005046                          CLR     -(SP)
        005610  156416  000107                   BISB    DRSEL+1(R4),(SP)
        005614  012746  003666                   MOV     #DRNM,-(SP)
        005620  016446  000104                   MOV     DCS(R4),-(SP)
        005624  012746  002367                   MOV     #MRLCS,-(SP)
        005630  013746  002270                   MOV     SECOND,-(SP)
        005634  013746  002272                   MOV     MINUTE,-(SP)
        005640  013746  002274                   MOV     HOUR,-(SP)
        005644  012746  002360                   MOV     #TIME,-(SP)
        005650  012746  006725                   MOV     #FMT10,-(SP)
        005654  012746  000011                   MOV     #11,-(SP)
        005660  010600                           MOV     SP,R0
        005662  104014                           EMT     C$PNTB
        005664  062706  000024                   ADD     #24,SP
    485 005670                                   PRINTB  #FMTDT,TDR(R4)
        005670  016446  000120                   MOV     TDR(R4),-(SP)
        005674  012746  007552                   MOV     #FMTDT,-(SP)
        005700  012746  000002                   MOV     #2,-(SP)
        005704  010600                           MOV     SP,R0
        005706  104014                           EMT     C$PNTB
        005710  062706  000006                   ADD     #6,SP
    486 005714  000207                           RTS     PC
    487
    488 005716  004737  005402          LINE3:   JSR     PC,LINE1
    489 005722                                   PRINTB  #FMT2,#CRLCS,BCSADR(R4),#CRLBA,@BBA(R4),#CRLDA,BDA(R4),#CRLMP,BMP(R4)
        005722  016446  000042                   MOV     BMP(R4),-(SP)
        005726  012746  002453                   MOV     #CRLMP,-(SP)
        005732  016446  000040                   MOV     BDA(R4),-(SP)
        005736  012746  002441                   MOV     #CRLDA,-(SP)
        005742  017446  000110                   MOV     @BBA(R4),-(SP)
        005746  012746  002427                   MOV     #CRLBA,-(SP)
        005752  016446  000046                   MOV     BCSADR(R4),-(SP)
        005756  012746  002377                   MOV     #CRLCS,-(SP)
        005762  012746  006302                   MOV     #FMT2,-(SP)
        005766  012746  000011                   MOV     #11,-(SP)
        005772  010600                           MOV     SP,R0
        005774  104014                           EMT     C$PNTB
        005776  062706  000024                   ADD     #24,SP
    490 006002                                   PRINTB  #FMT3,#CRLCS,E.CS,#CRLBA,E.BA,#CRLDA,E.DA,#CRLMP,E.MP
        006002  013746  002304                   MOV     E.MP,-(SP)
        006006  012746  002453                   MOV     #CRLMP,-(SP)
        006012  013746  002302                   MOV     E.DA,-(SP)
        006016  012746  002441                   MOV     #CRLDA,-(SP)
        006022  013746  002300                   MOV     E.BA,-(SP)
        006026  012746  002427                   MOV     #CRLBA,-(SP)
        006032  013746  002276                   MOV     E.CS,-(SP)
        006036  012746  002377                   MOV     #CRLCS,-(SP)
        006042  012746  006345                   MOV     #FMT3,-(SP)
        006046  012746  000011                   MOV     #11,-(SP)
        006052  010600                           MOV     SP,R0
        006054  104014                           EMT     C$PNTB
        006056  062706  000024                   ADD     #24,SP
    491 006062  013737  002302  002224           MOV     E.DA,TEMPO      ;GET ADDRESS TO PRINT
    492 006070  004537  006076                   JSR     R5,TELCYL       ;PRINT IT
    493 006074  000207                           RTS     PC              ;EXIT
    494
    495 006076  013737  002224  002164  TELCYL: MOV     TEMPO,CYL       ;GET THE ADDRESS
```

```
 496 006104  042737  000177  002164          BIC     #177,CYL        ;SAVE ONLY CYLINDER BITS
 497 006112  000337  002164                  SWAB    CYL
 498 006116  000241                          CLC
 499 006120  006137  002164                  ROL     CYL
 500 006124  103002                          BCC     1$
 501 006126  005237  002164                  INC     CYL
 502 006132  013737  002224  002170  1$:     MOV     TEMPO,SEC       ;GET SECTOR #
 503 006140  042737  177700  002170          BIC     #177700,SEC     ;SAVE ONLY THE SECTOR BITS
 504 006146  005037  002166                  CLR     SUR             ;INIT TO HEAD 0
 505 006152  032737  000100  002302          BIT     #100,E.DA       ;HEAD 1?
 506 006160  001405                          BEQ     2$              ;NO
 507 006162  005237  002166                  INC     SUR             ;YUP
 508 006166  042737  177776  002166          BIC     #177776,SUR
 509 006174                          2$:     PRINTB  #FMT3A,#DRVER,CYL,SUR,SEC
     006174  013746  002170                  MOV     SEC,-(SP)
     006200  013746  002166                  MOV     SUR,-(SP)
     006204  013746  002164                  MOV     CYL,-(SP)
     006210  012746  002716                  MOV     #DRVER,-(SP)
     006214  012746  006406                  MOV     #FMT3A,-(SP)
     006220  012746  000005                  MOV     #5,-(SP)
     006224  010600                          MOV     SP,R0
     006226  104014                          EMT     C$PNTB
     006230  062706  000014                  ADD     #14,SP
 510 006234  000205                          RTS     R5
 511
 512                                ;FORMAT STATMENTS
 513
 517
 518 006236      045     124     045  FMT1:    .ASCIZ  /%T%Z2%A:%Z2%A:%Z2/
 519 006260      045     124     045  FMT17:   .ASCIZ  /%T%O6%T%O1/
 520 006273      045     124     045  FMT1A:   .ASCIZ  /%T%T%N/
 521 006302      045     101     102  FMT2:    .ASCII  /%ABEFORE ERR%T%O6/
 522 006323      045     124     045  FMT2A:   .ASCIZ  /%T%O6%T%O6%T%O6%N/
 523 006345      045     101     101  FMT3:    .ASCIZ  /%AAT ERR    %T%O6%T%O6%T%O6%T%O6/
 524 006406      045     116     045  FMT3A:   .ASCIZ  /%N%T%A ADDR = CYL: %Z3%A. SUR: %O1%A SECT: %Z2%A.%N/
 525 006472      045     116     045  FMT4:    .ASCIZ  /%N%T%O6%T%O6%N%T%O6%T%O6%N/
 526 006525      045     124     045  FMT5:    .ASCIZ  /%T%O6%T%T%N/
 527 006541      045     124     045  FMT6:    .ASCIZ  /%T%O6%T%O6%A OR %O6%N/
 528 006567      045     124     045  FMT7:    .ASCIZ  /%T%A - %T%N/
 529 006603      045     104     066  FMT8:    .ASCIZ  /%D6%T%N/
 530 006613      045     124     045  FMT9:    .ASCIZ  /%T%Z2%A:%Z2%A:%Z2%T%O6%T%O1%N/
 531 006651      045     104     066  FMT9A:   .ASCIZ  /%D6%A. WORDS BAD OUT OF %D6%A. WORDS READ%N/
 532 006725      045     124     045  FMT10: .ASCIZ   /%T%Z2%A:%Z2%A:%Z2%T%O6%T%O1/
 533 006761      045     124     045  FMT10A: .ASCIZ  /%T%O6%T%O6%N%T%O6%T%O6%A AT BUS ADDRESS /
 534 007032      045     117     066  FMT10B: .ASCIZ  /%O6%N/
 535 007040      045     124     045  FMT11:   .ASCIZ  /%T%O2%T%O6%T%O3/
 536 007060      045     124     045  FMT12:   .ASCIZ  /%T%O6%N/
 537 007070      045     124     045  FMT13:   .ASCIZ  /%T%O6%T%O6%N/
 538 007105      045     124     045  FMT13D:  .ASCIZ  /%T%Z4%A NOW IS %Z4%N/
 539 007132      045     116     045  FMT14:   .ASCIZ  /%N%T%N/
 540 007141      045     117     066  FMT14A:  .ASCIZ  /%O6%A /
 541 007150      045     116     000  FMT14C:  .ASCIZ  /%N/
 542 007153      045     101     127  FMT14B:  .ASCIZ  ?%AWORD %D3%A. S/B %O6%A WAS %O6%N?
 543 007215      045     101     105  FMT15:   .ASCIZ  /%AERROR(S) SET:%T%N%ARECOVERY BEING ATTEMPTED/
 544 007273      045     101     116  FRMT16:  .ASCIZ  /%ANOT TESTING CS= %O6%A DR= %O1%N/
 545 007335      045     116     045  FMT18:   .ASCIZ  /%N%T/
 546 007342      045     116     045  FMTxS:   .ASCIZ  /%N%AXFER SIZE = %Z6%A. WORDS%N/
```

```
547 007401    045    116    045  FMTS1:   .ASCIZ  /%N%N%S10%A*** RL01-RL02 PERFORMANCE REPORT ***%N%N/
548 007464    045    101    052  FMTS1A:  .ASCIZ  /%A*** RUNNING%N/
549 007504    045    101    052  FMTS1B:  .ASCIZ  /%A*** DROPPED  %Z2%A:%Z2%N/
550 007537    045    124    045  FMTS2:   .ASCIZ  /%T%O5%O5%N/
551 007552    045    101    040  FMTDT:   .ASCIZ  /%A DRIVE TYPE = RL0%O1%N/
552 007603    045    101    124  FMTS2A:  .ASCIZ  /%ATOTAL SEEKS:     %D6%Z3%N%AWORDS READ:     %D6%Z4%Z4%N/
553 007672    045    101    127  FMTS2B:  .ASCIZ  /%AWORDS WRITTEN: %D6%Z4%Z4%N/
554 007727    045    116    045  FMTS3:   .ASCIZ  /%N%AERRORS%N%ADRV-ER:%D6%A   SEEK: %D6%A   TRACK: %D6%A   DATA:  %D6%N/
555 010036    045    101    110  FMTS3A:  .ASCIZ  /%AHARD:   %D6%A    SOFT: %D6%N/
556 010073    045    101    104  FMTS4:   .ASCIZ  /%ADCK:    %D6%A    HCRC: %D6%A   NXM:    %D6%A   HNF:    %D6%N/
557 010166    045    101    104  FMTS5:   .ASCIZ  /%ADLT:    %D6%A    OPI:  %D6%N%N/
558
562
563                              .EVEN
564
565 010226                                ENDMOD
566
567 010226                       BGNMOD   HPTCODE
568
569 010226                       BGNHW
    010226  000006                        .WORD   L10013-L$HW/2
570
571 010230  174400                        .WORD   174400
572 010232  000160                        .WORD   160
573 010234  000240                        .WORD   240
574 010236  000001                        .WORD   1
575 010240  000000                        .WORD   0
576 010242  000001                        .WORD   1
577
578 010244                       ENDHW
    010244                       L10013:
579
580 010244                                ENDMOD
581
582                              .SBTTL   SOFTWARE PARAMETERS
583
584 010244                       BGNMOD   SPTCODE
585
586 010244                       BGNSW
    010244  000040                        .WORD   L10014-L$SW/2
587
588 010246  000001               LIMIT:   .WORD   1                  ;RETRY LIMIT
589 010250  000003               ERLMT:   .WORD   3                  ;ERROR LIMIT
590 010252  000003               SELMT:   .WORD   3                  ;SEEK ERROR LIMIT
591 010254  060650               DALMT:   .WORD   25000.             ;DATA XFER LIMIT (*(10*3)) (BITS)
592 010256  023420               SKLMT:   .WORD   10000.             ;SEEK LIMIT
593 010260  001320               TYINT:   .WORD   720.               ;TIME INTERVAL BETW/ STATISTICAL REPORT (12 HRS)
594 010262  000020               CMRD:    .WORD   16.                ;WORDS TO COMPARE ON READ
595 010264  000003               DELMT:   .WORD   3                  ;ERRORS TO REPORT ON DATA COMPARE
596 010266  000000               XCHFLG:  .WORD   0                  ;CHANGE OTHER PARAMETERS
597 010270  002400               T.MXB:   .WORD   1280.              ;MAXIMUM R/W TRANSFER BUFFER
598 010272  000100               T.MXH:   .WORD   100                ;MAXIMUM HEAD SELECT
599 010274  000000               T.MNH:   .WORD   0                  ;MINIMUM HEAD SELECT
600 010276  177600               T.MXC:   .WORD   177600             ;MAXIMUM CYLINDER
601 010300  000000               T.MNC:   .WORD   0                  ;MINIMUM CYLINDER
602 010302  000000               T.MXS:   .WORD   0                  ;MAXIMUM START SECTOR
603 010304  000000               T.MNS:   .WORD   0                  ;MINIMUM START SECTOR
```

```
010306  000001                 T.DCK:   .WORD   1                        ;DATA DUMP ON DATA CHECK ERROR
010310  000001                 T.DRP:   .WORD   1                        ;DROP ON LIMIT REACHED
010312  000003                 T.MNB:   .WORD   3                        ;MINIMUM BUFFER TRANSFER SIZE
010314  000012                 SFLMT:   .WORD   10.                      ;SOFT ERROR LIMIT
010316  000000                 T.STA:   .WORD   0                        ;DROP DRIVE ON PERFORMANCE REACHED
010320  000003                 DRLMT:   .WORD   3                        ;DRIVE ERROR LIMIT
010322  000000                 T.ROF:   .WORD   0                        ;READ ONLY FLAG
010324  000001                 T.RAN:   .WORD   1                        ;RANDOM SELECT OF PATTERNS
010326  000004                 T.PAT:   .WORD   4                        ;ONLY ONE PATTERN  4 = WORST CASE
010330  000001                 T.SLT:   .WORD   1                        ;SEEK RETRY LIMIT
010332  000200                 T.CLT:   .WORD   128.                     ;NUMBER OF ERRORS ON DCK DUMP
010334  000000                 T.AUT:   .WORD   0                        ;AUTO ON START UP
010336  000000                 T.STIP:  .WORD   0                        ;RESTRICT BUFFER SIZE
010340  000001                 T.WCK:   .WORD   1                        ;DO WRITE CHECK
010342  000012                 T.DCD:   .WORD   10.
010344  000001                 T.ANS:   .WORD   1
010346                         ENDSW
010346                         L10014:
010346                                  ENDMOD
010346                         BGNMOD   DSPCODE
010346                         DISPATCH         1
010346  000001                          .WORD   1
010350  013152                          .WORD   T1
010352                                  ENDMOD
                               .SBTTL   STATISTIC CODE
010352                         BGNMOD   RPTCODE
010352                         BGNRPT
010352                                  PRINTS  #FMTS1                   ;PRINT STATISTICAL HEADER
010352  012746  007401                  MOV     #FMTS1,-(SP)
010356  012746  000001                  MOV     #1,-(SP)
010362  010600                          MOV     SP,R0
010364  104016                          EMT     C$PNTS
010366  062706  000004                  ADD     #4,SP
010372  010446                          MOV     R4,-(SP)                 ;SAVE PRESENT VALUE OF R4
010374  012704  026514                  MOV     #DRBUF,R4                ;START OF DRIVE BUFFER
010400  005764  000104         1$:      TST     DCS(R4)                  ;IS THERE A DRIVE?
010404  001402                          BEQ     2$                       ;NO, GET NEXT ONE
010406  004737  012442                  JSR     PC,REPORT                ;TYPE OUT SUMMARY
010412  062704  000126         2$:      ADD     #PRPOS+2,R4              ;NEXT DRIVE
010416  020427  027774                  CMP     R4,#ENDBUF               ;AT THE END?
010422  001366                          BNE     1$                       ;NO, TRY NEXT
010424  012604                          MOV     (SP)+,R4                 ;RESTORE R4
```

```
                                ENDRPT
                                L10015:
010426                                  EMT     C$RPT
                                
                                        ENDMOD
                                
                                .SBTTL  INITIALIZATION CODE
                                
                                BGNMOD  INITCODE                        ;START OF INITIALIZE CODE
                                
                                        BGNINIT
                                
                                        SETPRI  #340                    ;PRIORITY TO SEVEN
010430  012700  000340                  MOV     #340,R0
010434  104041                          EMT     C$SPRI
                                
                                        BRESET                          ;FOR LSI-11 CPU'S
010436  104033                          EMT     C$RESET
010440  005037  000050                  CLR     OPFLG
010444  005037  002356                  CLR     INCALL
010450  005037  002332                  CLR     STFLG
010454  005037  002334                  CLR     CNTFLG                  ;CLEAR CONT
                                        READEF  #EF.PWR
010460  012700  000034                  MOV     #EF.PWR,R0
010464  104050                          EMT     C$REFG
                                        BNCOMPLETE      3$
010466  103076                          BCC     3$
010470  005237  002326                  INC     PWRFLG                  ;INDICATE POWER FAIL
010474  012704  026514                  MOV     #DRBUF,R4
010500  012702  000001                  MOV     #1,R2
010504  130237  002136          11$:    BITB    R2,DRUT
010510  001446                          BEQ     13$
010512  016400  000106                  MOV     DRSEL(R4),R0
010516  052700  000200                  BIS     #200,R0
010522  010074  000104                  MOV     R0,@DCS(R4)
010526  012701  000074                  MOV     #60.,R1
010532  032774  000001  000104  12$:    BIT     #1,@DCS(R4)
010540  001014                          BNE     15$
                                        WAITMS  #10.
010542  012700  000012                  MOV     #10.,R0
010546  104026                          EMT     C$WTM
010550  005301                          DEC     R1
010552  001367                          BNE     12$
                                
010554  012737  003737  002132          MOV     #NOPWR,WHY
010562  004537  021502                  JSR     R5,DRDRV
010566  000137  010626                  JMP     13$
                                
010572  004537  022406          15$:    JSR     R5,ISDRST
010576  004537  023752                  JSR     R5,HDHOME
010602  005064  000056                  CLR     PRFLGS(R4)
010606  005064  000036                  CLR     RETRY(R4)
010612  005064  000076                  CLR     DOWCK(R4)
010616  005064  000052                  CLR     RTYPE(R4)
010622  005064  000114                  CLR     RSEEK(R4)
```

```
010626  062704  000126            13$:    ADD     #PRPOS+2,R4
010632  106302                            ASLB    R2
010634  103323                            BCC     11$
010636  005737  002314                    TST     SYSCLK
010642  001406                            BEQ     4$
010644                                    CLKON   #1
010644  012700  000001                    MOV     #1,R0
010650  104034                            EMT     C$KWON
010652                                    REQTIM  R0
010652  104045                            EMT     C$REQTIM
010654  010037  002266                    MOV     R0,LSTTIM
010660  000137  011712            4$:     JMP     POWER
010664                            3$:     READEF  #EF.CONTINUE            ;CONTINUE FROM CONSOLE?
010664  012700  000036                    MOV     #EF.CONTINUE,R0
010670  104050                            EMT     C$REFG
010672                                    BNCOMPLETE      1$              ;NO, CONTINUE W/ INIT CODE
010672  103004                            BCC     1$

010674  005237  002334                    INC     CNTFLG                  ;YES SET CONT FLAG, GO TO END OF INIT
010700  000137  011236                    JMP     END

010704  004537  025256            1$:     JSR     R5,CLEAR                ;CLEAR ALL DRIVE BUFFERS
010710  012737  176543  002144            MOV     #176543,HINUM           ;PRIME RANDOM GENERATOR
010716  012737  123456  002146            MOV     #123456,LONUM           ;   "        "        "
010724  012700  002176            2$:     MOV     #CNTLR1,R0              ;CLEAR FLAGS
010730  005020                    CLRDAT: CLR     (R0)+                   ;
010732  020027  002334                    CMP     R0,#STFLG+2             ;MASS CLEAR
010736  001374                            BNE     CLRDAT                  ;

010740  012704  026514                    MOV     #DRBUF,R4               ;SETUP UP DRIVE BUFFER POINTER
010744  012702  025446                    MOV     #BSEC0,R2               ;SETUP BAD SECTOR POINTER
010750  013703  002012                    MOV     L$UNIT,R3               ;GET NUMBER OF UNITS
010754  010337  002324                    MOV     R3,UUT                  ;SAVE L$UNIT
010760  005001                            CLR     R1                      ;INIT P-TABLE
010762  005703                    1$:     TST     R3                      ;ANY    P-TABLES LEFT?
010764  001524                            BEQ     END                     ;NO,GO TO END
010766                                    GPHARD  R1,R0                   ;GET A P-TABLE
010766  010100                            MOV     R1,R0
010770  104042                            EMT     C$GPHRD
010772                                    BNCOMPLETE      12$
010772  103112                            BCC     12$
010774  012037  002206                    MOV     (R0)+,BCSR              ;GET CSR
011000  012037  002210                    MOV     (R0)+,BVEC              ;GET VECTOR
011004  012037  002212                    MOV     (R0)+,BPRIOR            ;GET BPRIOR
011010  012037  002140                    MOV     (R0)+,T.DRIVE
011014  011037  002214                    MOV     (R0),BDRSEL             ;GET DRIVE
011020  005737  002176                    TST     CNTLR1                  ;DO WE HAVE CSR 1 YET?
011024  001011                            BNE     2$                      ;YES,THEN SEE IF IT'S IT
011026  013737  002212  002254            MOV     BPRIOR,PRIOR1
011034  013737  002206  002176            MOV     BCSR,CNTLR1             ;NO,MAKE THIS ONE CSR 1
011042  013737  002210  002250            MOV     BVEC,VECT1              ;MAKE THIS VECTOR VECT1
011050  023737  002206  002176    2$:     CMP     BCSR,CNTLR1             ;IS THIS CSR CNTLR1?
011056  001012                            BNE     5$                      ;NO,GO CHECK AGAINST #2
011060  023737  002210  002250            CMP     BVEC,VECT1              ;IS VECTOR PROPER?
011066  001050                            BNE     10$                     ;NO, REPORT ERROR
011070  012737  002316  002226            MOV     #BUF1,TEMP1             ;FIRST CONTROLLER/FIRST BUFFER
011076  004537  012206                    JSR     R5,FILINF               ;FILL BUFFER
```

```
011102  000450                          BR      11$                     ;GO GET NEXT P-TABLE
011104  005737  002200          5$:     TST     CNTLR2                  ;HAVE WE GOT CSR #2 YET?
011110  001015                          BNE     6$                      ;YES, CHECK THIS ONE AGAINST IT
011112  023737  002250  002206          CMP     VECT1,BCSR              ;IS THIS VECTOR SAME AS CNTLR1
011120  001433                          BEQ     10$                     ;IFSO, DON'T ALLOW IT
011122  013737  002206  002200          MOV     BCSR,CNTLR2             ;MAKE THIS ONE CSR 2
011130  013737  002210  002252          MOV     BVEC,VECT2              ;SETUP SECOND VECTOR
011136  013737  002212  002256          MOV     BPRIOR,PRIOR2
011144  023737  002206  002200  6$:     CMP     BCSR,CNTLR2             ;IS THIS CSR # 2?
011152  001016                          BNE     10$                     ;NO, WELL WE DON'T ALLOW 3
011154  023737  002210  002252          CMP     BVEC,VECT2              ;DOES IT HAVE PROPER VECTOR
011162  001012                          BNE     10$                     ;NO, GO REPORT ERROR
011164  023737  002252  002250          CMP     VECT2,VECT1             ;IS VECTOR OF FIRST EQUAL TO
011172  001406                          BEQ     10$                     ;VECTOR OF SECOND, YES REPORT ERROR
011174  012737  002320  002226          MOV     #BUF2,TEMP1             ;OTHER CNTLR/OTHER BUFFER
011202  004537  012206                  JSR     R5,FILINF               ;LOAD BUFFER
011206  000406                          BR      11$                     ;NEXT
011210                          10$:    ERRDF   160.,ILLEG,ERR10        ;BAD P-TABLE
011210  104462                          TRAP    T$ERCODE
011212  000240                          .WORD   160
011214  003626                          .WORD   ILLEG
011216  005234                          .WORD   ERR10
011220  005064  000104          12$:    CLR     DCS(R4)
011224  005201                  11$:    INC     R1                      ;POINT TO NEXT
011226  005303                          DEC     R3                      ;DOWN COUNT
011230  062702  000042                  ADD     #34.,R2                 ;NEXT BAD SECTOR FILE
011234  000652                          BR      1$                      ;DO WHILE


011236                          END:

011236  012737  177770  002142          MOV     #177770,SYSMSK          ;SETUP FOR EIGHT DRIVES
011244  023727  002324  000004          CMP     UUT,#4                  ;MORE THAN FOUR
011252  003012                          BGT     2$                      ;YES, THEN MASK IS OKAY
011254  052737  000004  002142          BIS     #4,SYSMSK               ;SETUP FOR FOUR DRIVES
011262  023727  002324  000002          CMP     UUT,#2                  ;MORE THAN TWO
011270  003003                          BGT     2$                      ;YES, IT'S OKAY
011272  052737  000002  002142          BIS     #2,SYSMSK               ;SET FOR ONE OR TWO
011300                          2$:     READEF  #EF.START               ;START COMMAND
011300  012700  000040                  MOV     #EF.START,R0
011304  104050                          EMT     C$REFG
011306                                  BNCOMPLETE      RESTART         ;NO, CHK RESTART
011306  103006                          BCC     RESTART
011310  005237  002332                  INC     STFLG                   ;SET START INDICATOR
011314  005037  002160                  CLR     WRINIT                  ;CLEAR THE WRITE INIT FLAG ON START
011320  005037  002174                  CLR     KILLDC                  ;CLEAR DATA COMP FLAG ON START ONLY

011324                          RESTART:
011324  005737  002334                  TST     CNTFLG                  ;CONTINUING
011330  001047                          BNE     3$                      ;YES GO TO 3$
011332  005737  002160                  TST     WRINIT                  ;IN PROCESS OF INITTING THE PACK?
011336  001420                          BEQ     11$                     ;NO
011340  005037  002160                  CLR     WRINIT                  ;YES - CLEAR THE FLAG
011344  005237  002174                  INC     KILLDC                  ;INHIBIT DATA COMPARES!
011350  005037  010262                  CLR     CMRD                    ;AND SET DAT COMPARE TO 0 WORDS
011354                                  PRINTF  #FMT18,#NORDDC          ;TELL OPR PACK NOT INITTED YET
011354  012746  004254                  MOV     #NORDDC,-(SP)
```

```
        011360  012746  007335                 MOV     #FMT18,-(SP)
        011364  012746  000002                 MOV     #2,-(SP)
        011370  010600                          MOV     SP,R0
        011372  104017                          EMT     C$PNTF
        011374  062706  000006                 ADD     #6,SP
   797
   798                                  ;LET'S CREATE INTERNAL BITMAP
   799
   800 011400  012701  000001          11$:    MOV     #1,R1                   ;BIT MASK
   801 011404  105037  002137                  CLRB    DRPRS                   ;CLEAR OUT DRIVES PRESENT
   802 011410  012704  026514                  MOV     #DRBUF,R4               ;START OF DRIVE BUFFERS
   803 011414  005764  000104          1$:     TST     DCS(R4)                 ;ANY CSR?
   804 011420  001402                          BEQ     2$                      ;NO, NO DRIVE THEN
   805 011422  150137  002137                  BISB    R1,DRPRS                ;INDICATE DRIVE IN BITMAP
   806 011426  006301                  2$:     ASL     R1                      ;NEXT POSITION
   807 011430  062704  000126                  ADD     #PRPOS+2,R4             ;NEXT DRIVE BUFFER
   808 011434  022704  027774                  CMP     #ENDBUF,R4              ;DONE
   809 011440  001365                          BNE     1$                      ;NO
   810
   811 011442  113737  002137  002136          MOVB    DRPRS,DRUT              ;SET UP DRIVES UNDER TEST
   812
   813 011450                           3$:
   814
   815 011450                                  SETVEC  VECT1,#INTR1,PRIOR1     ;SET CONTROLLER 1'S VECTOR
        011450  013746  002254                 MOV     PRIOR1,-(SP)
        011454  012746  015336                 MOV     #INTR1,-(SP)
        011460  013746  002250                 MOV     VECT1,-(SP)
        011464  012746  000003                 MOV     #3,-(SP)
        011470  104037                         EMT     C$SVEC
        011472  062706  000010                 ADD     #10,SP
   816
   817 011476  005737  002200                  TST     CNTLR2                  ;RUNNING TWO CONTROLLERS?
   818 011502  001413                          BEQ     4$                      ;NO
   819
   820 011504                                  SETVEC  VECT2,#INTR2,PRIOR2     ;YES SET CONTROLLER 2'S VECTOR
        011504  013746  002256                 MOV     PRIOR2,-(SP)
        011510  012746  015346                 MOV     #INTR2,-(SP)
        011514  013746  002252                 MOV     VECT2,-(SP)
        011520  012746  000003                 MOV     #3,-(SP)
        011524  104037                         EMT     C$SVEC
        011526  062706  000010                 ADD     #10,SP
   821
   822 011532  005737  002334          4$:     TST     CNTFLG                  ;CONTINUE?
   823 011536  001412                          BEQ     FINDBF                  ;NO, GO PAST RESTART OF CLOCK
   824
   825 011540  005737  002314                  TST     SYSCLK                  ;DO WE HAVE SYSTEM CLOCK
   826 011544  001462                          BEQ     POWER                   ;NO
   827
   828 011546                                  CLKON   #1                      ;TURN CLK ON
        011546  012700  000001                 MOV     #1,R0
        011552  104034                         EMT     C$KWON
   829 011554                                  REQTIM  R0                      ;REQUEST TIME
        011554  104045                         EMT     C$REQTIM
   830 011556  010037  002266                  MOV     R0,LSTTIM               ;MAKE IT PRESENT TIME
   831 011562  000453                          BR      POWER                   ;GO TO END
   832
   833
```

```
    834 011564  012703  000050          FINDBF: MOV     #40.,R3                 ;MAXIMUM SECTOR IS 40
    835 011570  005001                          CLR     R1                      ;START WC AT ZERO
    836 011572  005737  002200                  TST     CNTLR2                  ;TWO CONTROLLERS????
    837 011576  001402                          BEQ     1$                      ;NO, START WC AT 5120
    838 011600  012701  000024                  MOV     #20.,R1                 ;20 256 WORD BUFFERS (1 TRACKS WORTH)
    839 011604  062701  000024          1$:     ADD     #20.,R1                 ;WC TO 5120 PLUS 5120 (1 TRK EA CONTROLLER)
    840 011610                          2$:     BUFFER  R1,R2                   ;GET BUFFER IF AVAILABLE
        011610  010100                          MOV     R1,R0
        011612  104030                          EMT     C$BUFF
        011614  010002                          MOV     R0,R2
    841 011616                                  BCOMPLETE       4$              ;WAS AVAILABLE, THEN BR
        011616  103411                          BCS     4$
    842 011620  005737  002200                  TST     CNTLR2                  ;TWO CONTROLLERS???
    843 011624  001401                          BEQ     3$                      ;NO
    844 011626  005301                          DEC     R1                      ;ONE 256 WORD BUFFER LESS
    845 011630  005301                  3$:     DEC     R1                      ;ONE MORE LESS
    846 011632  162703  000002                  SUB     #2,R3                   ;256 WORDS = 2 SECTORS
    847 011636  001364                          BNE     2$                      ;IF NOT ZERO GO BACK
    848
    849 011640  000000                          HALT
    850
    851 011642  042701  177400          4$:     BIC     #177400,R1
    852 011646  000301                          SWAB    R1
    853 011650  010237  002316                  MOV     R2,BUF1                 ;GET BUFFER FOR FIRST CONTROLLER
    854 011654  005737  002200                  TST     CNTLR2                  ;TWO CONTROLLERS??
    855 011660  001404                          BEQ     5$                      ;NO
    856 011662  060102                          ADD     R1,R2                   ;SECOND'S BUFFER
    857 011664  010237  002320                  MOV     R2,BUF2
    858 011670  006201                          ASR     R1                      ;CORRECT WORD COUNT
    859 011672  010137  002322          5$:     MOV     R1,MAXWC                ;MAX WORD COUNT
    860
    861
    862
    863 011676                          7$:     CLKON   #1                      ;TURN CLOCK ON?
        011676  012700  000001                  MOV     #1,R0
        011702  104034                          EMT     C$KWON
    864 011704                                  BNCOMPLETE      POWER           ;WAS THERE A CLOCK?
        011704  103002                          BCC     POWER
    865
    866 011706  005237  002314                  INC     SYSCLK                  ;YES, SET FLAG FOR ONE!
    867
    868 011712                          POWER:
    869
    870
    871
    872 011712                                  ENDINIT
        011712                          L10016:
        011712  104011                          EMT     C$INIT
    873
    874 011714                                  ENDMOD
    875
    876 011714                          BGNMOD  CLNCODE
    877
    878
    879 011714                                  BGNCLN
    880
    881 011714                                  SETVEC  ERRVEC,#TRPHAN,#340
```

```
        011714  012746  000340                  MOV     #340,-(SP)
        011720  012746  012434                  MOV     #TRPHAN,-(SP)
        011724  013746  002346                  MOV     ERRVEC,-(SP)
        011730  012746  000003                  MOV     #3,-(SP)
        011734  104037                          EMT     C$SVEC
        011736  062706  000010                  ADD     #10,SP
    882 011742                                  SETPRI  #PRI00                  ;PRIORITY TO ZERO
        011742  012700  000000                  MOV     #PRI00,R0
        011746  104041                          EMT     C$SPRI
    883
    884 011750  032777  000200  170220  1$:     BIT     #CRDY,@CNTLR1           ;WAIT FOR CONTROLLER TO FINISH
    885 011756  001774                          BEQ     1$                      ;
    886 011760  042777  000100  170210          BIC     #INTEN,@CNTLR1          ;CLEAR INTERRUPT IF PENDING
    887 011766                                  CLRVEC  VECT1                   ;RELEASE VECTOR OF FIRST CONTROLLER
        011766  013700  002250                  MOV     VECT1,R0
        011772  104036                          EMT     C$CVEC
    888
    889 011774  005737  002200                  TST     CNTLR2                  ;TWO CONTROLLERS
    890 012000  001412                          BEQ     3$                      ;NO
    891
    892 012002  032777  000200  170170  2$:     BIT     #CRDY,@CNTLR2           ;WAIT FOR OTHER CONTROLLER TO FINISH
    893 012010  001774                          BEQ     2$                      ;
    894 012012  042777  000100  170160          BIC     #INTEN,@CNTLR2          ;CLEAR OUT INTERRUPT ENABLE
    895 012020                                  CLRVEC  VECT2                   ;YES, WELL RELEASE IT'S VECTOR
        012020  013700  002252                  MOV     VECT2,R0
        012024  104036                          EMT     C$CVEC
    896
    897 012026  005037  002356          3$:     CLR     INCALL
    898 012032  005037  002354                  CLR     OPCALL
    899 012036                                  CLRVEC  ERRVEC
        012036  013700  002346                  MOV     ERRVEC,R0
        012042  104036                          EMT     C$CVEC
    900 012044  005737  002314                  TST     SYSCLK
    901 012050  001401                          BEQ     4$
    902 012052                                  CLKOFF
        012052  104035                          EMT     C$KWOFF
    903 012054                          4$:     BRESET                          ;TAKE CARE OF LSI-11
        012054  104033                          EMT     C$RESET
    904 012056                                  ENDCLN
        012056                          L10017:
        012056  104012                          EMT     C$CLEAN
    905
    906 012060                                  ENDMOD
    907
    908
    909 012060                          BGNMOD  ADDCODE
    910
    911 012060                          BGNAU
    912
    913 012060  012704  026514                  MOV     #DRBUF,R4               ;START OF DRIVE BUFFERS
    914 012064  012701  000001                  MOV     #1,R1                   ;MASK TO FIND DRIVE
    915 012070  010002                          MOV     R0,R2                   ;SAVE WHICH TO FIND
    916 012072  005700                  1$:     TST     R0                      ;THIS ONE
    917 012074  001405                          BEQ     2$                      ;YES
    918 012076  062704  000126                  ADD     #PRPOS+2,R4             ;NEXT
    919 012102  006301                          ASL     R1                      ;NEXT MASK
    920 012104  005300                          DEC     R0
```

```
012106  000771                                  BR      1$
012110  150137  002136          2$:             BISB    R1,DRUT                 ;INSERT IN DRIVE UNDER TEST
012114                                          GPHARD  R2,R1
012114  010200                                  MOV     R2,R0
012116  104042                                  EMT     C$GPHRD
012120  010001                                  MOV     R0,R1
012122  011164  000104                          MOV     (R1),DCS(R4)
012126  012700  000100                          MOV     #SERNM1,R0              ;SETUP TO CLEAR STATS
012132  006200                                  ASR     R0
012134  005024                  4$:             CLR     (R4)+
012136  005300                                  DEC     R0
012140  001375                                  BNE     4$
012142                          5$:

012142                          ENDAU
012142                          L10020:
012142  104054                                  EMT     C$AU

012144                          ENDMOD

012144                          BGNMOD  DROPCODE

012144                          BGNDU

012144  005737  002356                          TST     INCALL
012150  001015                                  BNE     3$
012152  012704  026514                          MOV     #DRBUF,R4
012156  005700                  2$:             TST     R0
012160  001404                                  BEQ     1$
012162  005300                                  DEC     R0
012164  062704  000126                          ADD     #PRPOS+2,R4
012170  000772                                  BR      2$

012172  012737  003345  002132  1$:             MOV     #REQ,WHY
012200  004537  021476                          JSR     R5,ODRDRV
012204                          3$:


012204                          ENDDU
012204                          L10021:
012204  104055                                  EMT     C$DU

012206                          ENDMOD

                                .SBTTL  GLOBAL SUBROUTINES

012206                          BGNMOD  GLBSUB

                                ;
                                ;ROUTINE TO FILL DRIVE PARAMETER BUFFERS WITH INFO

012206  013764  002214  000106  FILINF: MOV     BDRSEL,DRSEL(R4)        ;SET DRIVE SELECT BITS
012214  022737  000001  002140          CMP     #1,T.DRIVE              ;DRIVE = RL01?
012222  001403                          BEQ     FILTD                   ;YES
012224  012737  000002  002140          MOV     #2,T.DRIVE              ;DRIVE IS AN RL02
012232  013764  002140  000120  FILTD:  MOV     T.DRIVE,TDR(R4)
012240  013764  002206  000104          MOV     BCSR,DCS(R4)            ;SET CSR
012246  013764  002226  000110          MOV     TEMP1,BBA(R4)           ;SET R/W BUFFER
```

```
    971 012254  010264  000112                  MOV     R2,BSECPT(R4)           ;SETUP BAD SECTOR POINTER
    972 012260  005737  010334                  TST     T.AUT                   ;DO WE AUTOSIZE?
    973 012264  001460                          BEQ     1$                      ;NO, SKIP
    974
    975 012266  005037  002330                  CLR     TRPFLG                  ;CLEAR  TRAP FLAG
    976 012272                                  SETVEC  ERRVEC,#TRPHAN,#340     ;SETUP TO CATCH TRAP
        012272  012746  000340                  MOV     #340,-(SP)
        012276  012746  012434                  MOV     #TRPHAN,-(SP)
        012302  013746  002346                  MOV     ERRVEC,-(SP)
        012306  012746  000003                  MOV     #3,-(SP)
        012312  104037                          EMT     C$SVEC
        012314  062706  000010                  ADD     #10,SP
    977 012320  005774  000104                  TST     @DCS(R4)
    978 012324  005737  002330                  TST     TRPFLG                  ;DID TRAP OCCUR
    979 012330  001012                          BNE     3$                      ;YES IGNORE DRIVE
    980 012332  016400  000106                  MOV     DRSEL(R4),R0            ;YES, FIND OUT IF DRIVE
    981 012336  052700  000200                  BIS     #200,R0                 ;HAS DRIVE READY POSTED
    982 012342  010074  000104                  MOV     R0,@DCS(R4)
    983 012346  032774  000001  000104          BIT     #1,@DCS(R4)             ;IS DRIVE READY HIGH?
    984 012354  001021                          BNE     2$                      ;YES, CHECK NEXT
    985
    986 012356                          3$:     PRINTF  #FRMT16,DCS(R4),<B,DRSEL+1(R4)>
        012356  005046                          CLR     -(SP)
        012360  156416  000107                  BISB    DRSEL+1(R4),(SP)
        012364  016446  000104                  MOV     DCS(R4),-(SP)
        012370  012746  007273                  MOV     #FRMT16,-(SP)
        012374  012746  000003                  MOV     #3,-(SP)
        012400  010600                          MOV     SP,R0
        012402  104017                          EMT     C$PNTF
        012404  062706  000010                  ADD     #10,SP
    987
    988 012410  005337  002324                  DEC     UUT                     ;ONE LESS DRIVE NOW
    989 012414  005064  000104                  CLR     DCS(R4)                 ;TAKE DRIVE OUT OF BUFFER
    990 012420                          2$:     CLRVEC  ERRVEC                  ;RELEASE THE VECTOR
        012420  013700  002346                  MOV     ERRVEC,R0
        012424  104036                          EMT     C$CVEC
    991 012426  062704  000126          1$:     ADD     #PRPOS+2,R4             ;UPDATE POINTER
    992 012432  000205                          RTS     R5
    993
    994 012434  005237  012434          TRPHAN: INC     TRPHAN
    995 012440  000002                          RTI
```

```
                                .SBTTL  REPORT ROUTINE
                                ;ROUTINE TO PRINT STATISTICAL REPORT OF DRIVE(S)

012442                          REPORT: PRINTS  #FMT10,#TIME,HOUR,MINUTE,SECOND,#MRLCS,DCS(R4),#DRNM,<B,DRSEL+1(R4)>
012442  005046                          CLR     -(SP)
012444  156416  000107                  BISB    DRSEL+1(R4),(SP)
012450  012746  003666                  MOV     #DRNM,-(SP)
012454  016446  000104                  MOV     DCS(R4),-(SP)
012460  012746  002367                  MOV     #MRLCS,-(SP)
012464  013746  002270                  MOV     SECOND,-(SP)
012470  013746  002272                  MOV     MINUTE,-(SP)
012474  013746  002274                  MOV     HOUR,-(SP)
012500  012746  002360                  MOV     #TIME,-(SP)
012504  012746  006725                  MOV     #FMT10,-(SP)
012510  012746  000011                  MOV     #11,-(SP)
012514  010600                          MOV     SP,R0
012516  104016                          EMT     C$PNTS
012520  062706  000024                  ADD     #24,SP
012524                                  PRINTS  #FMTDT,TDR(R4)
012524  016446  000120                  MOV     TDR(R4),-(SP)
012530  012746  007552                  MOV     #FMTDT,-(SP)
012534  012746  000002                  MOV     #2,-(SP)
012540  010600                          MOV     SP,R0
012542  104016                          EMT     C$PNTS
012544  062706  000006                  ADD     #6,SP
012550  005764  000070                  TST     DPHOUR(R4)              ;DO WE HAVE ANY DROPPED TIME
012554  001417                          BEQ     1$                      ;NO, THEN PRINT RUNNING

                                ;PRINT THE TIME THE DRIVE WAS DROPPED FROM TESTING

012556                                  PRINTS  #FMTS1B,<B,DPHOUR(R4)>,<B,DPMIN(R4)>
012556  005046                          CLR     -(SP)
012560  156416  000071                  BISB    DPMIN(R4),(SP)
012564  005046                          CLR     -(SP)
012566  156416  000070                  BISB    DPHOUR(R4),(SP)
012572  012746  007504                  MOV     #FMTS1B,-(SP)
012576  012746  000003                  MOV     #3,-(SP)
012602  010600                          MOV     SP,R0
012604  104016                          EMT     C$PNTS
012606  062706  000010                  ADD     #10,SP
012612  000410                          BR      2$

012614                          1$:     PRINTS  #FMTS1A         ;PRINT '*** RUNNING'
012614  012746  007464                  MOV     #FMTS1A,-(SP)
012620  012746  000001                  MOV     #1,-(SP)
012624  010600                          MOV     SP,R0
012626  104016                          EMT     C$PNTS
012630  062706  000004                  ADD     #4,SP

012634                          2$:     PRINTS  #FMTS2,#CART,SERNM2(R4),SERNM1(R4)
012634  016446  000100                  MOV     SERNM1(R4),-(SP)
012640  016446  000102                  MOV     SERNM2(R4),-(SP)
012644  012746  002476                  MOV     #CART,-(SP)
012650  012746  007537                  MOV     #FMTS2,-(SP)
012654  012746  000004                  MOV     #4,-(SP)
012660  010600                          MOV     SP,R0
012662  104016                          EMT     C$PNTS
```

```
                012664  062706  000012          ADD     #12,SP
     17 012670                                  PRINTS  #FMTS2A,SKCNT(R4),SKCNT1(R4),RXFR3(R4),RXFR2(R4),RXFR1(R4)
                012670  016446  000002          MOV     RXFR1(R4),-(SP)
                012674  016446  000004          MOV     RXFR2(R4),-(SP)
                012700  016446  000060          MOV     RXFR3(R4),-(SP)
                012704  016446  000054          MOV     SKCNT1(R4),-(SP)
                012710  016446  000000          MOV     SKCNT(R4),-(SP)
                012714  012746  007603          MOV     #FMTS2A,-(SP)
                012720  012746  000006          MOV     #6,-(SP)
                012724  010600                  MOV     SP,R0
                012726  104016                  EMT     C$PNTS
                012730  062706  000016          ADD     #16,SP
     18 012734                                  PRINTS  #FMTS2B,WXFR3(R4),WXFR2(R4),WXFR1(R4)
                012734  016446  000006          MOV     WXFR1(R4),-(SP)
                012740  016446  000010          MOV     WXFR2(R4),-(SP)
                012744  016446  000062          MOV     WXFR3(R4),-(SP)
                012750  012746  007672          MOV     #FMTS2B,-(SP)
                012754  012746  000004          MOV     #4,-(SP)
                012760  010600                  MOV     SP,R0
                012762  104016                  EMT     C$PNTS
                012764  062706  000012          ADD     #12,SP
     19 012770                                  PRINTS  #FMTS3,DERCNT(R4),SKECNT(R4),TRERR(R4),DATCER(R4)
                012770  016446  000074          MOV     DATCER(R4),-(SP)
                012774  016446  000072          MOV     TRERR(R4),-(SP)
                013000  016446  000016          MOV     SKECNT(R4),-(SP)
                013004  016446  000020          MOV     DERCNT(R4),-(SP)
                013010  012746  007727          MOV     #FMTS3,-(SP)
                013014  012746  000005          MOV     #5,-(SP)
                013020  010600                  MOV     SP,R0
                013022  104016                  EMT     C$PNTS
                013024  062706  000014          ADD     #14,SP
     20 013030                                  PRINTS  #FMTS3A,ERRCNT(R4),SFTCNT(R4)
                013030  016446  000014          MOV     SFTCNT(R4),-(SP)
                013034  016446  000012          MOV     ERRCNT(R4),-(SP)
                013040  012746  010036          MOV     #FMTS3A,-(SP)
                013044  012746  000003          MOV     #3,-(SP)
                013050  010600                  MOV     SP,R0
                013052  104016                  EMT     C$PNTS
                013054  062706  000010          ADD     #10,SP
     21 013060                                  PRINTS  #FMTS4,DCRCER(R4),HCRCER(R4),NXMCNT(R4),HNFERR(R4)
                013060  016446  000032          MOV     HNFERR(R4),-(SP)
                013064  016446  000034          MOV     NXMCNT(R4),-(SP)
                013070  016446  000024          MOV     HCRCER(R4),-(SP)
                013074  016446  000022          MOV     DCRCER(R4),-(SP)
                013100  012746  010073          MOV     #FMTS4,-(SP)
                013104  012746  000005          MOV     #5,-(SP)
                013110  010600                  MOV     SP,R0
                013112  104016                  EMT     C$PNTS
                013114  062706  000014          ADD     #14,SP
     22 013120                                  PRINTS  #FMTS5,DLTCNT(R4),OPICNT(R4)
                013120  016446  000030          MOV     OPICNT(R4),-(SP)
                013124  016446  000026          MOV     DLTCNT(R4),-(SP)
                013130  012746  010166          MOV     #FMTS5,-(SP)
                013134  012746  000003          MOV     #3,-(SP)
                013140  010600                  MOV     SP,R0
                013142  104016                  EMT     C$PNTS
                013144  062706  000010          ADD     #10,SP
```

```
      23 013150  000207                          RTS     PC
      24
      25
      26 013152                                  ENDMOD
      27
```

```
                                .SBTTL  PROGRAM MAIN LOOP
013152                          BGNTST
013152                          STARS
                                ;;*****************************************************************
                                ;PROGRAM WILL RANDOMLY PICK ONE OF THE DRIVES TO
                                ;PERFORM AN OPERATION.  WE WILL ALWAYS PICK ONE OF FOUR
                                ;OR EIGHT DRIVES (ONE OR TWO CONTROLLERS) "DRUT" WILL BE
                                ;CHECKED TO SEE IF DRIVE IS ON SYSTEM.  ONCE DRIVE IS PICKED
                                ;THEN A FUNCTION WILL BE SELECTED RANDOMLY FOR THAT
                                ;DRIVE.  FUNCTIONS OF CONTROLLER RESET, GET STATUS, SEEK, READ, WRITE
                                ;WILL BE SELECTED, EACH FUNCTION WILL HAVE IT'S OWN ROUTINE
                                ;TO GET PARAMETERS FOR THE DRIVE.
013152                          STARS
                                ;;*****************************************************************

013152                          MTEST:
013152  005737  002160                TST     WRINIT                  ;HERE AFTER PWR FAIL DURING WRITE
013156  001407                    BEQ     161$                    ;NO
013160  013704  002160            MOV     WRINIT,R4               ;YES - RESET R4
013164  013701  002162            MOV     WRPOS,R1                ;AND R1 POINTERS
013170  005237  002332            INC     STFLG                   ;FAKE OUT THE START FLAG
013174  000410                    BR      16$                     ;AND CONTINUE WRITE INIT CODE
013176  012704  026514    161$:   MOV     #DRBUF,R4               ;GET DRIVE BUFFERS
013202  012701  000001            MOV     #1,R1                   ;MASK
013206  010437  002160            MOV     R4,WRINIT               ;COPY THE R4 AND
013212  010137  002162            MOV     R1,WRPOS                ;POINTERS

013216  130137  002136    16$:    BITB    R1,DRUT                 ;DRIVE UNDFR TEST
013222  001442                    BEQ     15$                     ;NO

013224  012774  000200  000104        MOV     #200,@DCS(R4)           ;CHECK IF DRIVE THERE
013232  056474  000106  000104        BIS     DRSEL(R4),@DCS(R4)
013240  012700  000000                MOV     #0.,R0                  ;STALL
013244  005300            13$:    DEC     R0
013246  001376                    BNE     13$
013250  032774  000001  000104        BIT     #DRDY,@DCS(R4)          ;WAIT FOR DRIVE TO BECOME 'READY'
013256  001006                    BNE     14$                     ;AFTER THE HEADS HOME COMMAND

013260  012737  002526  002132        MOV     #DNRDY,WHY
013266  004537  021502                JSR     R5,DRDRV
013272  000416                    BR      15$

013274  004537  020636    14$:    JSR     R5,RDBDSC               ;GO GET BAD SECTORS
013300  005064  000056            CLR     PRFLGS(R4)
013304  005064  000114            CLR     RSEEK(R4)
013310  005764  000122            TST     WRIPG(R4)               ;SEE IF WRITE IN PROGRESS FLAG SET
013314  001003                    BNE     99$                     ;JUMP IF SET
013316  005737  002332            TST     STFLG
013322  001402                    BEQ     15$

013324  004537  022562    99$:    JSR     R5,WRPACK

013330  062704  000126    15$:    ADD     #PRPOS+2,R4             ;NEXT DRIVE
013334  010437  002160            MOV     R4,WRINIT               ;SAVE CURRENT R4 POINTER
013340  006337  002162            ASL     WRPOS                   ;AND SHIFT COPY OF R1 POINTER
013344  106301                    ASLB    R1                      ;DONE?
013346  103323                    BCC     16$                     ;NO GO FOR NEXT ONE
```

```
                                ;HERE WHEN ALL FINISHED WITH THE WRITE INIT CODE
013350  005037  002160                CLR     WRINIT                  ;CLEAR THE WRITE INIT FLAG
013354                      12$:      PRINTF  #FMT14,#MSTART
013354  012746  004216                MOV     #MSTART,-(SP)
013360  012746  007132                MOV     #FMT14,-(SP)
013364  012746  000002                MOV     #2,-(SP)
013370  010600                        MOV     SP,R0
013372  104017                        EMT     C$PNTF
013374  062706  000006                ADD     #6,SP
013400                              SETPRI  #0              ;PRIORITY TO ZERO
013400  012700  000000                MOV     #0,R0
013404  104041                        EMT     C$SPRI
013406  004537  022464      MAIN:     JSR     R5,RAND         ;GET A DRIVE?(LUN)
013412  013702  002146                MOV     LONUM,R2        ;GET THE SELECTED DRIVE (LUN)
013416  043702  002142      PEROTH:   BIC     SYSMSK,R2       ;MASK TO DRIVES ON SYSTEM
013422  012701  000001                MOV     #1,R1           ;LET'S SEE IF DRIVE IS THERE
013426  005702              1$:       TST     R2              ;HAVE WE GOT PROPER MASK YET
013430  001403                        BEQ     2$              ;YES, GO TO 2$
013432  006301                        ASL     R1              ;NO, SHIFT FOR NEXT DRIVE
013434  005302                        DEC     R2              ;DECREMENT DRIVE NUMBER
013436  000773                        BR      1$              ;GO CHECK NEW DRIVE NUMBER
013440  105737  002136      2$:       TSTB    DRUT            ;ANY DRIVES ON LINE
013444  001005                        BNE     5$              ;YES, CHECK

013446                              ERRSF   170.,NODRIV     ;NO DRIVES
013446  104421                      TRAP    T$ERCODE
013450  000252                      .WORD   170
013452  003654                      .WORD   NODRIV

013454  000137  026506              JMP     ENDOFPROGRAM

013460  130137  002136      5$:       BITB    R1,DRUT         ;IS THIS DRIVE PRESENT?
013464  001750                        BEQ     MAIN            ;NO, GO BACK TRY AGAIN
013466  010137  002134                MOV     R1,TSTDRV       ;COPY UNIT UNDER TEST FOR LATER CHECK

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

013472  004537  023630                JSR     R5,GETSYS       ;GET PRESENT TIME OF SYSTEM
013476  023737  002264  010260        CMP     INTERVAL,TYINT  ;TIME TO PRINT REPORT
013504  002403                        BLT     6$              ;NO, PERFORM FUNCTION
013506  005037  002264                CLR     INTERVAL        ;YES, START INTERVAL OVER

013512                              DORPT                   ;PRINT STATISTICAL REPORT
013512  104024                      EMT     C$DRPT

013514  012704  026514      6$:       MOV     #DRBUF,R4       ;GET START OF DRIVE BUFFERS
013520  013702  002146                MOV     LONUM,R2        ;GET RANDOM DRIVE BACK (LUN)
013524  043702  002142                BIC     SYSMSK,R2       ;MASK TO SYSTEM SYS
013530  005702              3$:       TST     R2              ;DO WE HAVE BUFFER FOR THAT DRIVE
013532  001404                        BEQ     4$              ;YES, GO CHECK IT'S CONTROLLER
013534  062704  000126                ADD     #PRPOS+2,R4     ;NO, UPDATE FOR NEXT BUFFER
013540  005302                        DEC     R2              ;DOWN COUNT DRIVE NUMBER (LUN)
013542  000772                        BR      3$              ;GO BACK AND CHECK FOR FOUND
013544  032774  000200  000104  4$:    BIT     #BIT7,@DCS(R4)  ;CONTROLLER ASSOCIATED WITH DRIVE
```

```
    101 013552  001715                        BEQ     MAIN                ;BUSY
    102 013554  032774  000100  000104        BIT     #BIT6,@DCS(R4)      ;INTERRUPT BEEN SERVICED?
    103 013562  001311                        BNE     MAIN                ;NO - WAIT FOR THE INTERRUPT
    104
    105                                 ;WE CAN NOW PROCEED IN GETTING A FUNCTION AND RELATED DATA
    106                                 ;FOR THE DRIVE RANDOMLY.  R4 HAS DRIVE BUFFER POINTER
    107
    108 013564                          TAGX:
    109 013564  005737  010310                  TST     T.DRP                       ;DROP ON ERROR LIMITS REACHED?
    110 013570  001456                          BEQ     GETFNC                      ;NO
    111 013572  026437  000012  010250          CMP     ERRCNT(R4),ERLMT            ;HARD REACHED?
    112 013600  103404                          BLO     9$
    113 013602  012737  003157  002132          MOV     #ERLMTM,WHY
    114 013610  000442                          BR      11$
    115 013612  026437  000014  010314  9$:     CMP     SFTCNT(R4),SFLMT            ;SOFT REACHED?
    116 013620  103404                          BLO     10$
    117 013622  012737  003222  002132          MOV     #SFEMSG,WHY
    118 013630  000432                          BR      11$
    119 013632  026437  000074  010342  10$:    CMP     DATCER(R4),T.DCD
    120 013640  103404                          BLO     110$
    121 013642  012737  003244  002132          MOV     #DCDMSG,WHY
    122 013650  000422                          BR      11$
    123 013652  016401  000016          110$:   MOV     SKECNT(R4),R1
    124 013656  066401  000072                  ADD     TRERR(R4),R1
    125 013662  020137  010252                  CMP     R1,SELMT
    126 013666  103404                          BLO     7$
    127 013670  012737  003201  002132          MOV     #SERLMT,WHY
    128 013676  000407                          BR      11$
    129 013700  026437  000020  010320  7$:     CMP     DERCNT(R4),DRLMT            ;DRIVE ERROR REACHED?
    130 013706  103407                          BLO     GETFNC                      ;NO - TIME TO DO SOMETHING
    131 013710  012737  003267  002132          MOV     #DERMSG,WHY
    132
    133 013716  004537  021502          11$:    JSR     R5,DRDRV                    ;DROP THIS DRIVE!!!
    134 013722  000137  013406                  JMP     MAIN                        ;GO GET ANOTHER
```

```
                                        ;HERE TO GET A 'STRING' FUNCTION - LIST OF COMMANDS TO ISSUE

013726                                  GETFNC:
013726  005737  010316                8$:     TST     T.STA           ;DO WE WISH TO DROP ON OPR LIMITS
013732  001422                                BEQ     98$             ;NO

013734  026437  000000  010256                CMP     SKCNT(R4),SKLMT ;PAST THE SEEK LIMIT??
013742  103416                                BLO     98$             ;NO, THEN GO TEST
013744  016400  000060                        MOV     RXFR3(R4),R0    ;GET READ COUNT
013750  066400  000062                        ADD     WXFR3(R4),R0    ;ADD IN WRITE COUNT
013754  020037  010254                        CMP     R0,DALMT        ;LIMIT  REACHED??
013760  103407                                BLO     98$             ;NO, THEN GO TEST
013762  012737  003446  002132                MOV     #SOPLMT,WHY
013770  004537  021502                        JSR     R5,DRDRV        ;DROP THE DRIVE
013774  000137  013406                        JMP     MAIN            ;GO FOR ANOTHER DRIVE

014000  004537  022464                98$:    JSR     R5,RAND         ;GET A RANDOM FUNCTION INDEX NUMBER
                                                                        ;0 & 7 ARE NOT LEGIT
014004  013702  002146                        MOV     LONUM,R2        ;GET IT
014010  042702  177770                        BIC     #177770,R2      ;MASK TO 0-7
014014  001001                                BNE     6$              ;IF 0, MAKE 1
014016  005202                                INC     R2
014020  022702  000007                6$:     CMP     #7,R2           ;IS IT 7?
014024  001001                                BNE     5$              ;IF 7, MAKE 6
014026  005302                                DEC     R2
014030  006302                        5$:     ASL     R2              ;SHIFT LEFT (X2)
014032  000172  020620                        JMP     @LIST(R2)       ;GO TO FUNCTION ROUTINE

014036                                  STARS
                                        ;;*****************************************************************
                                        ;SKWRT -- ISSUE:
                                        ;       SEEK TO A CYLINDER
                                        ;       WRITE DATA
                                        ;       WRITE CHECK
014036                                  STARS
                                        ;;*****************************************************************

014036  004537  014420                SKWRT:  JSR     R5,SKFNC        ;RANDOM SEEK LOAD
014042  004537  014166                        JSR     R5,OPROK        ;WAIT TILL DONE
014046  000240                                NOP                     ;FOR DEBUGGING
014050  000240                                NOP
014052  004537  015062                        JSR     R5,WRTFNC       ;WRITE DATA LOAD
014056  004537  014166                        JSR     R5,OPROK
014062  004537  014356                        JSR     R5,WRTCKF       ;WRITE CHECK LOAD
014066  004537  014166                        JSR     R5,OPROK
014072  000137  013406                        JMP     MAIN            ;GET NEXT COMMAND
```

```
      1 014076                          STARS
                                        ;;*****************************************************************
      2                                 ;SKRD -- ISSUE:
      3                                 ;       RANDOM SEEK TO A CYLINDER
      4                                 ;       READ DATA
      5 014076                          STARS
                                        ;;*****************************************************************
      6
      7 014076  004537  014420          SKRD:   JSR     R5,SKFNC        ;LOAD SEEK
      8 014102  004537  014166                  JSR     R5,OPROK
      9 014106  000240                          NOP
     10 014110  000240                          NOP
     11 014112  004537  015132                  JSR     R5,RDDFNC       ;LOAD READ DATA CMD
     12 014116  004537  014166                  JSR     R5,OPROK
     13 014122  000137  013406                  JMP     MAIN            ;GET THE NEXT COMMAND
     14
     15 014126                          STARS
                                        ;;*****************************************************************
     16                                 ;SKRDRD -- ROUTINE TO DO:
     17                                 ;       SEEK TO A CYLINDER
     18                                 ;       READ (AND COMPARE DATA)
     19                                 ;       READ (AGAIN)
     20 014126                          STARS
                                        ;;*****************************************************************
     21
     22 014126  004537  014420          SKRDRD: JSR     R5,SKFNC        ;LOAD SEEK
     23 014132  004537  014166                  JSR     R5,OPROK
     24 014136  000240                          NOP
     25 014140  000240                          NOP
     26 014142  004537  015132          JSR     R5,RDDFNC       ;LOAD READ
     27 014146  004537  014166          JSR     R5,OPROK
     28 014152  004537  015132          JSR     R5,RDDFNC       ;LOAD READ
     29 014156  004537  014166          JSR     R5,OPROK
     30 014162  000137  013406          JMP     MAIN            ;EXIT
```

```
014166                              STARS
                                    ;;*****************************************************************
                                    ;OPROK -- ROUTINE TO ISSUE THE FUNCTION AND WAIT FOR 'READY'...IF AN
                                    ;        ERROR RETRY IS NEEDED - THEN ISSUE THE FUNCTION AGAIN.
014166                              STARS
                                    ;;*****************************************************************

014166  004537  015226              OPROK:  JSR     R5,LDFUNC       ;ISSUE THE FUNCTION
014172  004537  022322                      JSR     R5,WTRDY        ;WAIT TILL READY
014176  133737  002134  002136              BITB    TSTDRV,DRUT     ;DRIVE STILL AVAILABLE?
014204  001003                              BNE     1$              ;YUP - CONTINUE
014206  005726                              TST     (SP)+           ;NO - FIX THE STACK
014210  000137  013406                      JMP     MAIN            ;BACK TO THE MAIN LOOP - FORCED EXIT FROM
                                                                    ;THE STRING FUNCTION
014214  005764  000036              1$:     TST     RETRY(R4)       ;NEED TO RETRY FUNCTION?
014220  001403                              BEQ     3$              ;NO
014222  004537  015174              2$:     JSR     R5,ISSUE        ;YES - ISSUE THE FUNCTION AGAIN
014226  000757                              BR      OPROK           ;AND DO IT
014230  005764  000114              3$:     TST     RSEEK(R4)       ;SEEK RETRY?
014234  001403                              BEQ     4$              ;NO - EXIT NOW
014236  004537  014420                      JSR     R5,SKFNC        ;DO A SEEK AGAIN
014242  000751                              BR      OPROK           ;ISSUE & EXECUTE THE SEEK
014244  000205                      4$:     RTS     R5              ;EXIT
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;SKRH -- ISSUE:
                                        ;       RANDOM SEEK
                                        ;       READ HEADERS
                                        ;       READ DATA W/NO HDR CMP
                                        ;       GET STATUS
                                        STARS
                                        ;;*****************************************************************

014246  004537  014420        SKRH:   JSR     R5,SKFNC        ;LOAD SEEK
014252  004537  015226                JSR     R5,LDFUNC       ;ISSUE
014256  004537  022322                JSR     R5,WTRDY
014262  000240                        NOP
014264  000240                        NOP
014266  004537  015050                JSR     R5,RDHFNC       ;LOAD READ HDRS
014272  004537  015226                JSR     R5,LDFUNC       ;ISSUE
014276  004537  022322                JSR     R5,WTRDY
014302  004537  014336                JSR     R5,RDNHC        ;LOAD READ W/NO HDRS
014306  004537  015226                JSR     R5,LDFUNC       ;ISSUE
014312  004537  022322                JSR     R5,WTRDY
014316  004537  014400                JSR     R5,GSTFNC       ;LOAD GET STATUS
014322  004537  015226                JSR     R5,LDFUNC       ;ISSUE
014326  004537  022322                JSR     R5,WTRDY
014332  000137  013406                JMP     MAIN            ;GET THE NEXT COMMAND

                                        STARS
                                        ;;*****************************************************************
                                        ;READ DATA W/NO HDR COMPARE
                                        STARS
                                        ;;*****************************************************************

014336  012764  177600  000042  RDNHC:  MOV     #-128.,BMP(R4)  ;SET FOR A 1 SECTOR READ
014344  012764  000016  000044          MOV     #16,FUNC(R4)    ;LOAD THE COMMAND
014352  000137  015174                JMP     ISSUE           ;PROCESS IT

                                        STARS
                                        ;;*****************************************************************
                                        ;WRTCKF - WRITE CHECK FUNCTION
                                        STARS
                                        ;;*****************************************************************

014356  005737  010322        WRTCKF: TST     T.ROF           ;READ ONLY SET?
014362  001401                        BEQ     1$              ;NO - DO THE WRITE-CHECK FUNCTION
014364  000205                        RTS     R5              ;YES - EXIT NOW

014366  012764  000002  000044  1$:     MOV     #WRCHK,FUNC(R4) ;SAVE CMD
014374  000137  015174                JMP     ISSUE           ;PROCESS IT
```

```
                                        .SBTTL  ROUTINES TO SETUP AND ISSUE GET STATUS & SEEK
014400                                  STARS
                                        ;;*****************************************************************
                                        ;GET STATUS FUNCTION
014400                                  STARS
                                        ;;*****************************************************************

014400  012764  000004  000044  GSTFNC: MOV     #GSTAT,FUNC(R4) ;LOAD GET STATUS
014406  012764  000003  000040          MOV     #GSBIT,BDA(R4)  ;SET GSBIT IN COMMAND WORD
014414  000137  015174                  JMP     ISSUE           ;GO ISSUE FUNCTION

014420                                  STARS
                                        ;;*****************************************************************
                                        ;SEEK FUNCTION
014420                                  STARS
                                        ;;*****************************************************************

                                        ;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

014420  005764  000114          SKFNC:  TST     RSEEK(R4)       ;TRYING TO RECOVER
014424  001003                          BNE     10$             ;YES - DO IT
014426  005764  000036                  TST     RETRY(R4)       ;RECOVERY FROM A 'DRIVE' ERROR?
014432  001411                          BEQ     98$             ;NO - NORMAL SEEK REQUIRED
014434  016401  000050          10$:    MOV     LSTHDR(R4),R1   ;YES SET UP FOR RESEEK
014440  016402  000124                  MOV     PRPOS(R4),R2    ;TO CYLINDER
014444  042701  000100                  BIC     #100,R1         ;HEAD SET IN LATER
014450  042702  000100                  BIC     #100,R2         ;
014454  000546                          BR      4$              ;SKIP RANDOM PART
014456  004537  022464          98$:    JSR     R5,RAND         ;GET A RANDOM NUMBER
014462  013702  002146                  MOV     LONUM,R2        ;GET THE RANDOM NUMBER
014466  043702  002156                  BIC     SMSK,R2         ;LEAVE CYL AND HEAD
014472  020264  000124                  CMP     R2,PRPOS(R4)    ;ON THAT TRACK ALREADY
014476  001767                          BEQ     98$             ;YES - RESELECT

014500  022764  000001  000120  980$:   CMP     #1,TDR(R4)      ;THIS DRIVE AN RL01?
014506  001006                          BNE     981$            ;NO - MUST BE AN RL02
014510  042702  100000                  BIC     #BIT15,R2       ;KILL UPPER BIT OF CYL ADDRESS
014514  022702  077700                  CMP     #077700,R2      ;POINTING TO THE BAD SEC FILE?
014520  001007                          BNE     96$             ;NO - PROCEED
014522  000403                          BR      982$            ;YUP - CORRECT THE POSITION
014524  022702  177700          981$:   CMP     #177700,R2      ;RL02 BAD SECTOR FILE?
014530  001003                          BNE     96$             ;NO - PROCEED
014532  000240          982$:   NOP                             ;TRAP
014534  042702  000100                  BIC     #HEAD,R2        ;POINT TO HEAD 0 LAST TRACK

014540  010237  002220          96$:    MOV     R2,CHKSEC       ;SAVE THE ADDRESS FOR THE BAD SEC FILE CHECK
014544  004537  025356                  JSR     R5,CKBDTK       ;SEE IF THIS ADDR IN BAD SECTOR FILE
014550  005737  002216                  TST     HDRFND          ;WAS IT?
014554  001340                          BNE     98$             ;YES - RESELECT THE ADDRESS
```

```
014556  005003                          90$:    CLR     R3
014560  010200                                  MOV     R2,R0                   ;COPY ADDRESS - NO SECTOR YET
014562  042700  177677                          BIC     #177677,R0              ;LEAVE ONLY HEAD
014566  023737  010276  010300                  CMP     T.MXC,T.MNC             ;MIN AND MAX CYLINDERS THE SAME
014574  001011                                  BNE     95$                     ;NO, BRANCH AND STAY IN LIMITS
014576  013702  010276                          MOV     T.MXC,R2                ;MAKE CYLINDER MAX/MIN
014602  022764  000001  000120                  CMP     #1,TDR(R4)              ;DRIVE = RL01?
014610  001031                                  BNE     92$                     ;NO
014612  042702  100000                          BIC     #BIT15,R2               ;FORCE CYL TO PROPER LIMIT
014616  000426                                  BR      92$                     ;GO CALCULATE DIFF AND SEEK
014620  042702  000100                  95$:    BIC     #HEAD,R2                ;STRIP OUT H.S. BIT
014624  023702  010276                  94$:    CMP     T.MXC,R2                ;IS ADDRESS LESS/EQUAL THAN MAX
014630  103010                                  BHIS    93$                     ;YES, CHECK LOW END
014632  005203                                  INC     R3                      ;BUMP A TALLY COUNTER
014634  020327  000012                          CMP     R3,#10.                 ;IF CAN'T FIND ADDRESS IN 10 TIMES THEN RESELECT
014640  001706                                  BEQ     98$                     ;RESELECT
014642  006202                                  ASR     R2                      ;HALF IT AND CHECK AGAIN
014644  062702  000200                  91$:    ADD     #BIT7,R2                ;JUST TO MAKE NON ZERO
014650  000763                                  BR      95$                     ;GO BACK AND CHECK AGAIN
014652  023702  010300                  93$:    CMP     T.MNC,R2                ;IS MIN GREATER/EQUAL THAN ADDRESS
014656  101406                                  BLOS    92$                     ;YES, CALCULATE DIFF AND SEEK
014660  005203                                  INC     R3
014662  020327  000012                          CMP     R3,#10.                 ;TIME TO RESELECT?
014666  001673                                  BEQ     98$                     ;YUP - DO IT NOW
014670  006302                                  ASL     R2                      ;NO, DOUBLE IT
014672  000764                                  BR      91$                     ;GO CHECK MAX/MIN AGAIN
014674  016401  000124                  92$:    MOV     PRPOS(R4),R1            ;GET PRESENT DISK POSITION
014700  042701  000177                          BIC     #177,R1
014704  022764  000001  000120                  CMP     #1,TDR(R4)              ;RL01=1
014712  001002                                  BNE     25$                     ;BRANCH...MUST BE RL02
014714  042702  100000                          BIC     #BIT15,R2               ;CLEAR THE HIGH BIT FOR RL02 CYL #
014720  016464  000124  000050          25$:    MOV     PRPOS(R4),LSTHDR(R4)
014726  010264  000124                          MOV     R2,PRPOS(R4)            ;NEW HEADER AFTER SEEK
014732  050064  000124                          BIS     R0,PRPOS(R4)            ;SET IN RANDOM HEAD GOTTEN
014736  023737  010272  010274                  CMP     T.MXH,T.MNH             ;MIN AND MAX HEAD SELECT THE SAME
014744  001012                                  BNE     4$                      ;NO, THEN WE CAN USE BOTH SURFACES
014746  005737  010272                          TST     T.MXH                   ;WHICH IS OUR SURFACE FOR USE
014752  001004                                  BNE     97$                     ;TOP SURFACE BRANCH
014754  042764  000100  000124                  BIC     #HEAD,PRPOS(R4)         ;LOWER SURFACE ONLY
014762  000403                                  BR      4$
014764  052764  000100  000124          97$:    BIS     #HEAD,PRPOS(R4)         ;TOP SURFACE ONLY
```

```
      1 014772                                  STARS
                                                ;;*****************************************************************
      2                                         ;CALCULATE THE DIFFERENCE WORD AND STORE IT IN BDA
      3 014772                                  STARS
                                                ;;*****************************************************************
      4
      5 014772  1601G2                  4$:     SUB     R1,R2           ;SUBTRACT PRESENT FROM NEXT
      6 014774  103002                          BCC     1$              ;IF POSITIVE RESULT GO TO 1$
      7 014776  005402                          NEG     R2              ;NEG RESULT, NEGATE IT
      8 015000  000402                          BR      2$              ;GO SET DIRECTION OUT
      9 015002  052702  000004         1$:     BIS     #SIGN,R2        ;DIRECTION OUT, MARKER
     10 015006  052702  000001         2$:     BIS     #MK,R2          ;MARKER BIT
     11 015012  032764  000100  000124         BIT     #HEAD,PRPOS(R4) ;WHICH SURFACE SELECTED?
     12 015020  001402                          BEQ     3$              ;TOP, THEN 3$
     13 015022  052702  000020                  BIS     #SKHS,R2        ;BOTTOM SET HEAD BIT
     14 015026  010264  000040         3$:     MOV     R2,BDA(R4)      ;MOVE DIFFERENCE WORD TO DA
     15 015032  010264  000066                  MOV     R2,DIFWD(R4)    ;LOAD DIFFERENCE WORD
     16 015036  012764  000006  000044         MOV     #SEEK,FUNC(R4)  ;LOAD SEEK
     17 015044  000137  015174                  JMP     ISSUE
```

```
                                        .SBTTL  ROUTINE TO LOAD READ HEADER AND ISSUE IT.

015050  012764  000010  000044  RDHFNC: MOV     #RDHDR,FUNC(R4) ;LOAD READ HEADER
015056  000137  015174                  JMP     ISSUE

                                        .SBTTL  ROUTINE TO LOAD WRITE DATA COMMAND

015062  005737  010322          WRTFNC: TST     T.ROF           ;READ ONLY
015066  001021                          BNE     RDDFNC          ;YES
015070  004537  024046                  JSR     R5,GWCDA        ;GET WORD COUNT,DA
015074  005737  010262                  TST     CMRD            ;COMPARE DATA ON A READ?
015100  001404                          BEQ     1$              ;NO - SO DON'T GEN A WRITE BUFFER
015102  005237  002172                  INC     REGEN           ;YES - SET THE GENERATE DATA FLAG
                                        ;
                                        ;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
                                        ;

015106  004537  020352                  JSR     R5,WRBUF        ;WRITE BUFFER INTO MEMORY
015112  012764  000012  000044  1$:     MOV     #WRITE,FUNC(R4) ;LOAD WRITE
015120  012764  000001  000122          MOV     #1,WRIPG(R4)    ;SET THE WRITE IN PROGRESS FLAG
015126  000137  015174                  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

015132  004537  024046          RDDFNC: JSR     R5,GWCDA        ;GET WORD COUNT, DA
015136  005737  010262                  TST     CMRD            ;GOING TO COMPARE DATA AFTER READING?
015142  001407                          BEQ     2$              ;NO - SO SKIP THE CLEAR BUFFER CODE
015144  016402  000042                  MOV     BMP(R4),R2      ;CLEAR OUT BUFFER AREA
015150  017401  000110                  MOV     @BBA(R4),R1     ;SO WE KNOW READ
015154  005021                  1$:     CLR     (R1)+           ;WORKED!!
015156  005202                          INC     R2
015160  001375                          BNE     1$
015162  012764  000014  000044  2$:     MOV     #READ,FUNC(R4)  ;LOAD READ
015170  000137  015174                  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"
                                        ;
015174  026437  000104  002176  ISSUE:  CMP     DCS(R4),CNTLR1  ;DRIVE ON CONTROLLER 1?
015202  001003                          BNE     1$              ;NO, ASSUME ON CONTROLLER 2
015204  010437  002202                  MOV     R4,LSTDR1       ;PUT BUFFER POINTER IN 1
015210  000402                          BR      2$              ;SKIP OVER NEXT INSTRUCTION
015212  010437  002204          1$:     MOV     R4,LSTDR2       ;PUT BUFFER POINTER IN 2
015216  052764  000100  000044  2$:     BIS     #INTEN,FUNC(R4) ;ALLOW INTERRUPTS
015224  000205                          RTS     R5              ;EXIT

                                        .SBTTL  ROUTINE TO LOAD FUNCTION
015226                                  STARS
                                        ;;**********************************************************************
                                        ;CALL   JSR     R5,LDFUNC
                                        ;ALL INFORMATION MUST BE SET UP IN DRIVE BUFFER
                                        ;R4 HAS POINTER TO BUFFER
015226                                  STARS
                                        ;;**********************************************************************

015226  016403  000104          LDFUNC: MOV     DCS(R4),R3      ;GET CSR FOR DRIVE
015232  032713  000200                  BIT     #BIT7,(R3)      ;CAN WE ISSUE COMMAND?
015236  001003                          BNE     1$              ;YES, GO ISSUE COMMAND

015240                                  ERRSF   200.,PRGER      ;THIS ERROR SHOULD NEVER PRINT
015240  104421                          TRAP    T$ERCODE
015242  000310                          .WORD   200
015244  002567                          .WORD   PRGER

015246  017463  000110  000002  1$:     MOV     @BBA(R4),BA(R3) ;LOAD BUS ADDRESS REGISTER
015254  016463  000040  000004          MOV     BDA(R4),DA(R3)  ;LOAD DISK ADDRESS REGISTER
015262  016463  000042  000006          MOV     BMP(R4),MP(R3)  ;LOAD MULTI-PURPOSE REGISTER
015270  016464  000044  000046          MOV     FUNC(R4),BCSADR(R4)     ;GET FUNCTION
015276  056464  000106  000046          BIS     DRSEL(R4),BCSADR(R4)    ;SET DRIVE SELECT BITS
015304  052764  000201  000046          BIS     #CRDY!DRDY,BCSADR(R4)   ;SET CRDY:DRDY IN IMAGE
015312  042764  002000  000046          BIC     #OPI,BCSADR(R4)         ;WE'RE CLEAR BIT 10 FOR DRIVE 7-4 (OKAY?)
015320  016463  000046  000000          MOV     BCSADR(R4),CS(R3)       ;LOAD CSR
015326  042763  000200  000000          BIC     #CRDY,CS(R3)            ;ISSUE FUNCTION
015334  000205                          RTS     R5              ;EXIT
```

```
                        .SBTTL  INTERRUPT SERVICE ROUTINES
                        
015336                  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
                        ; 7-    READ W/NO HDR COMPARE - 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
                        ;
                        ;
```

```
 1 015336  010446                    INTR1:  MOV     R4,-(SP)        ;SAVE PRESENT R4 VALUE
 2 015340  013704  002202                    MOV     LSTDR1,R4       ;GET THE DRIVE BUFFER OF INTERRUPTING DRIVE
 3 015344  000403                            BR      SAVE            ;GO SAVE R0-R3
 4 015346  010446                    INTR2:  MOV     R4,-(SP)        ;SAVE PRESENT R4 VALUE
 5 015350  013704  002204                    MOV     LSTDR2,R4       ;GET THE DRIVE BUFFER OF INTERRUPTING DRIVE
 6 015354  013746  002276            SAVE:   MOV     E.CS,-(SP)
 7 015360  013746  002300                    MOV     E.BA,-(SP)
 8 015364  013746  002302                    MOV     E.DA,-(SP)
 9 015370  013746  002304                    MOV     E.MP,-(SP)
10 015374  013746  002306                    MOV     E.MP1,-(SP)
11 015400  013746  002310                    MOV     E.MP2,-(SP)
12 015404  013746  002220                    MOV     CHKSEC,-(SP)
13 015410  013746  002216                    MOV     HDRFND,-(SP)
14 015414  013746  002226                    MOV     TEMP1,-(SP)
15 015420  013746  002132                    MOV     WHY,-(SP)
16 015424  013746  002354                    MOV     OPCALL,-(SP)
17 015430  013746  002356                    MOV     INCALL,-(SP)
18 015434  010346                          MOV     R3,-(SP)        ;SAVE R3
19 015436  010246                          MOV     R2,-(SP)        ;R2
20 015440  010146                          MOV     R1,-(SP)        ;R1
21 015442  010046                          MOV     R0,-(SP)        ;R0
22 015444  005064  000122                  CLR     WRIPG(R4)       ;CLEAR THE WRITE IN PROGRESS FLAG
23 015450  016403  000104                  MOV     DCS(R4),R3      ;GET CSR FOR INTERRUPT
24 015454  016337  000000  002276          MOV     CS(R3),E.CS     ;SAVE ALL REGISTERS NOW!!
25 015462  016337  000002  002300          MOV     BA(R3),E.BA
26 015470  016337  000004  002302          MOV     DA(R3),E.DA
27 015476  016337  000006  002304          MOV     MP(R3),E.MP
28 015504  016337  000006  002306          MOV     MP(R3),E.MP1
29 015512  016337  000006  002310          MOV     MP(R3),E.MP2
30 015520  005737  002276                  TST     E.CS            ;ANY ERRORS?
31 015524  100402                          BMI     1$              ;YES, GO SOLVE ERROR MYSTERY
32 015526  000137  016652                  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.

015532 013764 002302 000064 1$:     MOV     E.DA,LSTDA(R4)  ;SAVE DA FOR SOFT ERROR PRINT
015540 032737 040000 002276         BIT     #DERR,E.CS      ;DRIVE ERROR?
015546 001402                       BEQ     2$              ;NO, CONTINUE
015550 000137 017636                JMP     CKDERR          ;YES, GO CHECK DRIVE ERROR
015554 032737 000001 002276 2$:     BIT     #DRDY,E.CS      ;DRIVE READY THERE
015562 001017                       BNE     23$             ;YES, CONTINUE CHECKING
015564 004537 022372                JSR     R5,GETDST       ;NO,GET DRIVE STATUS
015570 042701 000100                BIC     #100,R1         ;GET RID OF HEAD
015574 020127 000034                CMP     R1,#34          ;ALLOW ONLY SEEK TRACKING STATE
015600 001410                       BEQ     23$             ;WAS 34 SKIP ERROR

015602 005264 000012                INC     ERRCNT(R4)      ;INDICATE HARD ERROR
015606                              ERRDF   1000.,NORDY,ERR9
015606 104462                       TRAP    T$ERCODE
015610 001750                       .WORD   1000
015612 002541                       .WORD   NORDY
015614 005170                       .WORD   ERR9

015616 000137 017472                JMP     EXIT1

015622 032737 020000 002276 23$:    BIT     #NXM,E.CS       ;NON-EXISTANT MEMORY?
015630 001407                       BEQ     3$              ;NO, KEEP CHECKING
015632 012764 004204 000052         MOV     #MTNXM,RTYPE(R4)  ;ERROR MESSAGE
015640 005264 000034                INC     NXMCNT(R4)      ;LOG ERROR
015644 000137 016256                JMP     111$            ;CHECK RETRY, EXIT BACK

015650 032737 014000 002276 3$:     BIT     #BIT12!BIT11,E.CS  ;QUALIFING BITS SET?
015656 001020                       BNE     5$              ;YES, CAN'T BE OPI ALONE

015660 032737 002000 002276         BIT     #OPI,E.CS       ;OPI SET?
015666 001006                       BNE     4$              ;YES, CONTINUE

015670                              ERRSF   10.,UDERR,ERR1  ;WE HAVE AN UNDIAGNOSABLE CONDITION, ONLY COMPOSITE SET
015670 104461                       TRAP    T$ERCODE
015672 000012                       .WORD   10
015674 002644                       .WORD   UDERR
015676 004456                       .WORD   ERR1
015700                      33$:    BREAK
015700 104022                       EMT     C$BRK
015702 000776                       BR      33$

015704 012764 004177 000052 4$:     MOV     #MTOPI,RTYPE(R4);SET UP FOR "OPI" PRINT
015712 005264 000030                INC     OPICNT(R4)      ;LOG ERROR
015716 000557                       BR      111$            ;CHECK RETRY EXIT BACK

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

015720 032737 002000 002276 5$:     BIT     #OPI,E.CS       ;OPI QUALIFIER SET?
015726 001060                       BNE     7$              ;YES, THEN IT'S HCRC OR HNF
```

```
                                ;IT'S NOW DOWN TO DLT OR DCRC
                                
015730  032737  010000  002276        BIT     #DLT,E.CS       ;DATA LATE?
015736  001406                      BEQ     6$              ;NO, MUST BE DATA CRC
015740  012764  004172  000052        MOV     #MTDLT,RTYPE(R4);SET UP FOR "DLT"PRINT
015746  005264  000026                INC     DLTCNT(R4)      ;LOG ERROR
015752  000541                      BR      111$            ;CHECK RETRY, EXIT
                                
015754  013737  002302  002220  6$:   MOV     E.DA,CHKSEC     ;SET UP SECTOR TO LOOK FOR
015762  005364  000064              DEC     LSTDA(R4)       ;DOWN COUNT FOR PRINT OUT
015766  005337  002220              DEC     CHKSEC          ;DOWN COUNT FOR LOOP UP
015772  004537  025300              JSR     R5,CKBDSC       ;CHECK BAD SECTOR LIST
015776  005737  002216              TST     HDRFND          ;WAS HEADER THERE?
016002  001117                      BNE     110$            ;IGNORE ERROR, RETURN
016004  005264  000022        117$: INC     DCRCER(R4)      ;ACCOUNT FOR ERROR
016010  012764  004165  000052        MOV     #MTDCRC,RTYPE(R4);SET UP FOR "DCRC" PRINT
016016  022764  000102  000044        CMP     #INTEN!WRCHK,FUNC(R4)
016024  001001                      BNE     118$
016026  000513                      BR      111$
                                
016030  005737  010306        118$: TST     T.DCK           ;DUMP BUFFER?
016034  001510                      BEQ     111$            ;NO, EXIT
016036                              PRINTF  #FMT14,#DMPDCK
016036  012746  003122                MOV     #DMPDCK,-(SP)
016042  012746  007132                MOV     #FMT14,-(SP)
016046  012746  000002                MOV     #2,-(SP)
016052  010600                      MOV     SP,R0
016054  104017                      EMT     C$PNTF
016056  062706  000006                ADD     #6,SP
016062  004537  024354              JSR     R5,DMPBUF       ;DUMP BUFFER
                                
016066  000473                      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.
                                
016070  032737  010000  002276  7$:   BIT     #HNF,E.CS       ;HEADER NOT FOUND SET?
016076  001470                      BEQ     112$            ;NO IT MUST BE HCRC
016100  012701  000051              MOV     #41.,R1         ;ALLOW FOURTY READ HEADERS TO
016104  004537  022406        8$:   JSR     R5,ISDRST
016110  016402  000106              MOV     DRSEL(R4),R2    ;FIND CYLINDER
016114  052702  000010              BIS     #RDHDR,R2       ;READ HEADER
016120  016403  000104              MOV     DCS(R4),R3
016124  010263  000000              MOV     R2,CS(R3)       ;ISSUE READ HEADER
016130  004537  022322              JSR     R5,WTRDY        ;WAIT
016134  005301                      DEC     R1              ;DONE 40 OF THESE?
016136  001424                      BEQ     9$              ;YES, GIVE UP WE DON'T HAVE ALL DAY!
016140  005763  000000              TST     CS(R3)          ;IS ERROR SET?
016144  100757                      BMI     8$              ;YES, GO DO IT AGAIN
                                
016146  016301  000006              MOV     MP(R3),R1       ;GET HEADER
016152  010137  002312              MOV     R1,C.HDR        ;SAVE FOR ERROR REPORTING
```

```
016156  043701  002156                  BIC     SMSK,R1          ;MASK OUT SECTOR BITS
016162  020164  000124                  CMP     R1,PRPOS(R4)     ;IS CYLINDER HEAD CORRECT?
016166  001415                          BEQ     10$              ;YES, GO CHECK BAD SECTOR LIST

016170  005264  000072                  INC     TRERR(R4)
016174                                  ERRHRD  20.,TRACK,ERR2   ;TRACKING DRIFT ERROR
016174  104463                          TRAP    T$ERCODE
016176  000024                          .WORD   20
016200  003142                          .WORD   TRACK
016202  004464                          .WORD   ERR2

016204  000137  017166                  JMP     SKRETRY          ;FIX TRACKING ERROR

016210                          9$:     ERRHRD  30.,EXHAUS,ERR1  ;WE CAN'T FIND GOOD HEADER ON THIS TRACK
016210  104463                          TRAP    T$ERCODE
016212  000036                          .WORD   30
016214  002630                          .WORD   EXHAUS
016216  004456                          .WORD   ERR1

016220  000410                          BR      110$

016222  013737  002302  002220  10$:    MOV     E.DA,CHKSEC
016230  004537  025356                  JSR     R5,CKBDTK        ;GO CHECK BAD SECTOR FILE
016234  005737  002216                  TST     HDRFND           ;WAS IT THERE
016240  001401                          BEQ     11$              ;NO, LOG IT EXIT
016242  000577                  110$:   BR      GOERRX           ;YES IGNORE ERROR

016244  005264  000032          11$:    INC     HNFERR(R4)       ;LOG IT
016250  012764  004152  000052          MOV     #MTHNF,RTYPE(R4);SET UP FOR "HNF" PRINT
016256  000573                  111$:   BR      GOFIN            ;EXIT


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

016260  022764  000110  000044  112$:   CMP     #INTEN!RDHDR,FUNC(R4)   ;READ HEADER?
016266  001417                          BEQ     13$              ;YES, GO FIND OUT MORE ABOUT IT
                                                                 ;NO, IT MUST BE R/W
016270  013737  002302  002220          MOV     E.DA,CHKSEC
016276  004537  025300                  JSR     R5,CKBDSC        ;BAD SECTOR SEARCH
016302  005737  002216                  TST     HDRFND           ;WAS OUR DA THERE?
016306  001401                          BEQ     12$              ;NO, MUST BE LEGIT ERROR
016310  000554                          BR      GOERRX           ;YES, IGNORE ERROR

016312  005264  000024          12$:    INC     HCRCER(R4)       ;LOG ERROR
016316  012764  004157  000052          MOV     #MTHCRC,RTYPE(R4)
016324  000550                          BR      GOFIN

016326  017401  000110          13$:    MOV     @BBA(R4),R1      ;USE IT'S BUFFER TO STORE HDRS
016332  012737  000050  002226          MOV     #40.,TEMP1       ;40 CONSECUTIVE HEADERS
016340  012702  000010          14$:    MOV     #RDHDR,R2        ;READ HEADER
016344  056402  000106                  BIS     DRSEL(R4),R2     ;
```

```
                        016350  016403  000104                  MOV     DCS(R4),R3      ;
                        016354  010263  000000                  MOV     R2,CS(R3)       ;
                        016360  004537  022322                  JSR     R5,WTRDY        ;WAIT FOR READY
                        016364  016321  000000                  MOV     CS(R3),(R1)+    ;READ ALL REGISTERS
                        016370  016321  000006                  MOV     MP(R3),(R1)+    ;
                        016374  016321  000006                  MOV     MP(R3),(R1)+    ;
                        016400  016321  000006                  MOV     MP(R3),(R1)+    ;
                        016404  005337  002226                  DEC     TEMP1           ;DONE 40 YET?
                        016410  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.

                        016412  017402  000110          99$:    MOV     @BBA(R4),R2     ;GET BUFFER START
                        016416  012701  000050                  MOV     #40.,R1         ;FOURTY HEADERS
                        016422  032712  002000          15$:    BIT     #OPI,(R2)       ;IS OPI SET IN CS
                        016426  001403                          BEQ     16$             ;NO, WELL CAN'T BE HCRC
                        016430  032712  004000                  BIT     #HCRC,(R2)      ;INSURE HCRC W/OPI
                        016434  001005                          BNE     17$             ;FOUND GO SEE IF IT COMPARES
                        016436  062702  000010          16$:    ADD     #10,R2          ;NEXT CS IMAGE
                        016442  005301                          DEC     R1              ;DONE 40
                        016444  001366                          BNE     15$
                        016446  000721                          BR      12$

                        016450  020274  000110          17$:    CMP     R2,@BBA(R4)     ;IS HEADER FIRST ONE?
                        016454  001046                          BNE     21$             ;NO, READ PREVIOUS HEADER
                                                                                        ;YES, WE'LL HAVE TO GO THRU
                                                                                        ;AND CHECK OTHERS BEFORE WE
                                                                                        ;CAN SAFELY CALCULATE
                                                                                        ;"SUPPOSED" BAD SECTOR
                        016456  017401  000110                  MOV     @BBA(R4),R1
                        016462  012703  000001                  MOV     #1,R3
                        016466  062701  000010          18$:    ADD     #10,R1
                        016472  032711  002000                  BIT     #OPI,(R1)
                        016476  001416                          BEQ     19$
                        016500  032711  004000                  BIT     #HCRC,(R1)
                        016504  001413                          BEQ     19$
                        016506  005203                          INC     R3
                        016510  022703  000017                  CMP     #15.,R3
                        016514  001364                          BNE     18$

                        016516  012737  003525  002132          MOV     #MBDMSC,WHY     ;DROP DRIVE DUE TO
                        016524  004537  021502                  JSR     R5,DRDRV        ;MORE THAN 16 BAD SECTORS
                        016530  000137  017472                  JMP     EXIT1

                        016534  005012          19$:    CLR     (R2)            ;CLEAR THIS CS
                        016536  062701  000002                  ADD     #2,R1           ;GET IT'S HEADER ADDRESS
                        016542  011102                          MOV     (R1),R2         ;GET HEADER
                        016544  010201                          MOV     R2,R1           ;SAVE HEADER
                        016546  042702  177700                  BIC     #177700,R2      ;MASK   ONLY SECTOR
                        016552  160301                          SUB     R3,R1           ;BACK UP TO SECTOR WHICH IS BAD
                        016554  100402                          BMI     20$             ;IF MINUS DO MAGIC
                        016556  160302                          SUB     R3,R2           ;NO THEN SUBTRACT IS LEGAL
                        016560  000421                          BR      22$             ;BRANCH TO CHECK FILE
```

```
206 016562  160302                20$:   SUB     R3,R2           ;THIS SUB PRODUCES WRONG ANSWER
207 016564  062702  000050              ADD     #50,R2          ;FIX IT UP
208 016570  000415                      BR      22$             ;GO CHECK FILE
209
210 016572  005012                21$:   CLR     (R2)            ;CLEAR THIS CS OUT
211 016574  162702  000006              SUB     #6,R2           ;GET PREVIOUS HEADER
212 016600  011201                      MOV     (R2),   R1
213 016602  005201                      INC     R1
214 016604  010102                      MOV     R1,R2
215 016606  042701  177700              BIC     #177700,R1
216 016612  022701  000050              CMP     #40.,R1
217 016616  002402                      BLT     22$
218 016620  162702  000050              SUB     #40.,R2
219 016624  010237  002220        22$:   MOV     R2,CHKSEC
220 016630  004537  025300              JSR     R5,CKBDSC
221 016634  005737  002216              TST     HDRFND
222 016640  001664                      BEQ     99$
223 016642  000137  017476        GOERRX: JMP     ERREX
224
225
226 016646  000137  017600        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.
                        ;

016652  016401  000044  CHKFNC: MOV     FUNC(R4),R1     ;GET FUNCTION OF DRIVE
016656  006201                  ASR     R1              ;ALIGN THE FUNCTION CODE
016660  042701  000040          BIC     #40,R1          ;WIPE OUT INT. ENAB (SHIFTED)
016664  005301                  DEC     R1              ;WRITE CHECK??
016666  001002                  BNE     2$              ;NO, BRANCH
016670  000137  017026          JMP     AFWRCK          ;FUNCTION #1

016674  005301          2$:     DEC     R1              ;GET STATUS?
016676  001564                  BEQ     AGSTAT          ;BRANCH IF SO...FUNCTION #2
016700  005301                  DEC     R1              ;SEEK?
016702  001420                  BEQ     ASEEK           ;BRANCH IF SO...FUNCTION #3
016704  005301                  DEC     R1              ;RDHDR?
016706  001477                  BEQ     ARDHDR          ;BRANCH IF SO...FUNCTION #4
016710  005301                  DEC     R1              ;WRITE?
016712  001002                  BNE     1$              ;NO, BRANCH
016714  000137  017354          JMP     AWRITE          ;FUNCTION #5
016720  005301          1$:     DEC     R1              ;READ?
016722  001431                  BEQ     AFREAD          ;BRANCH IF SO...FUNCTION #6
016724  005301                  DEC     R1              ;READ W/NO HDR COMPARE?
016726  001437                  BEQ     AFWRCK          ;YES - TREAT AS IF WRITE CHECK

016730                          ERRSF   210.,PRGER      ;SHOULD NEVER GET HERE!!!
016730  104421                  TRAP    T$ERCODE
016732  000322                  .WORD   210
016734  002567                  .WORD   PRGER
016736  000000                  HALT
016740  000137  017440  XEXIT:  JMP     EXIT
```

```
                                        .SBTTL  SEEK INTERRUPT SERVICE
                                
 016744  052764  000001  000056 ASEEK:  BIS     #SKDON,PRFLGS(R4)  ;SET SEEK VERIFY NEEDED
 016752  005064  000114         CLR     RSEEK(R4)          ;CLEAR THE RETRY FLAG
 016756  005264  000054         INC     SKCNT1(R4)         ;INCREMENT COUNT
 016762  026427  000054  001750 CMP     SKCNT1(R4),#1000.  ;10(3) REACHED
 016770  002404                 BLT     99$                ;NO, EXIT
 016772  005264  000000         INC     SKCNT(R4)          ;YES, BUMP THOUSANDS
 016776  005064  000054         CLR     SKCNT1(R4)
 017002  000137  017440 99$:    JMP     EXIT
                                
                                        .SBTTL  READ INTERRUPT SERVICE
                                
 017006  042764  000001  000056 AFREAD: BIC     #SKDON,PRFLGS(R4)       ;CLEAR THE SEEK VERIFY FLAG
 017014                         SETPRI  #340
 017014  012700  000340         MOV     #340,R0
 017020  104041                 EMT     C$SPRI
 017022  004537  021724         JSR     R5,CKDATA          ;CHECK DATA
                                
 017026  016401  000042 AFWRCK: MOV     BMP(R4),R1         ;BUMP UP XFER COUNT
 017032  005401                 NEG     R1                 ;MAKE POSITIVE
 017034  060164  000002         ADD     R1,RXFR1(R4)       ;ADD THE BITS
 017040  022764  023420  000002 CMP     #10000.,RXFR1(R4)  ;10(8) REACHED YET
 017046  101016                 BHI     2$                 ;NO, EXIT
 017050  005264  000004         INC     RXFR2(R4)          ;BUMP 10(10)
 017054  162764  023420  000002 SUB     #10000.,RXFR1(R4)  ;START 10(8) AT 0
 017062  022764  023420  000004 CMP     #10000.,RXFR2(R4)  ;10(10) REACHED YET
 017070  101005                 BHI     2$                 ;NO, EXIT
 017072  005264  000060         INC     RXFR3(R4)          ;YES BUMP 65K 10(10)
 017076  162764  023420  000004 SUB     #10000.,RXFR2(R4)  ;MAKE 10(10) 0
 017104  000555         2$:     BR      EXIT               ;EXIT
                                
                                        .SBTTL  READ HEADER INTERRUPT SERVICE
                                
 017106  013701  002304 ARDHDR: MOV     E.MP,R1            ;GET HEADER
 017112  043701  002156         BIC     SMSK,R1            ;MASK OUT SECTOR BITS
 017116  026401  000124         CMP     PRPOS(R4),R1       ;IS HEADER CORRECT?
 017122  001442                 BEQ     1$                 ;YES, CONTINUE
                                
 017124  032764  000001  000056 BIT     #SKDON,PRFLGS(R4)  ;IS THIS MIS-SEEK OR TRACKING ERROR
 017132  001407                 BEQ     2$                 ;BRANCH IF TRACKING
                                
 017134  005264  000016         INC     SKECNT(R4)         ;ACCOUNT FOR SEEK ERROR
 017140                         ERRHRD  50.,MSKER,ERR2
 017140  104463                 TRAP    T$ERCODE
 017142  000062                 .WORD   50
 017144  002666                 .WORD   MSKER
 017146  004464                 .WORD   ERR2
 017150  000406                 BR      3$                 ;BRANCH AROUND TRACKING ERROR REPORT
                                
 017152  005264  000072 2$:     INC     TRERR(R4)          ;ACCOUNT FOR TRACKING ERROR
 017156                         ERRHRD  55.,TRACK,ERR2     ;TRACKING ERROR
 017156  104463                 TRAP    T$ERCODE
 017160  000067                 .WORD   55
 017162  003142                 .WORD   TRACK
 017164  004464                 .WORD   ERR2
```

```
                017166                  SKRETRY=.

017166  005264  000114        3$:       INC     RSEEK(R4)  ;SET RETRY IN PROGRESS
017172  026437  000114  010330          CMP     RSEEK(R4),T.SLT ;RETRY EXHAUSTED?????
017200  101405                          BLOS    4$              ;NO, THEN RETRY

017202                                  ERRHRD  333.,SEXHAU,ERR2
017202  104463                          TRAP    T$ERCODE
017204  000515                          .WORD   333
017206  003360                          .WORD   SEXHAU
017210  004464                          .WORD   ERR2
017212  000406                          BR      1$

017214  010164  000050        4$:       MOV     R1,LSTHDR(R4)   ;SET UP RETRY
017220  042764  000001  000056          BIC     #SKDON,PRFLGS(R4)  ;ALLOW SEEK
017226  000504                          BR      EXIT            ;EXIT
017230  042764  000001  000056  1$:     BIC     #SKDON,PRFLGS(R4)  ;SET VERIFICATION DONE
017236  005064  000114                  CLR     RSEEK(R4)
017242  010164  000124                  MOV     R1,PRPOS(R4)    ;MAKE THIS HEADER PRESENT POSITION
017246  000474                          BR      EXIT            ;EXIT

                                .SBTTL  GET STATUS INTERRUPT SERVICE

017250  013701  002304        AGSTAT: MOV     E.MP,R1         ;GET STATUS
017254  042701  000100                  BIC     #100,R1         ;CLEAR OUT HEAD SELECT
017260  005737  010322                  TST     T.ROF           ;READ ONLY
017264  001402                          BEQ     2$
017266  042701  020000                  BIC     #WL,R1
017272  032701  177400        2$:       BIT     #177400,R1      ;ANY BITS WRONG
017276  001406                          BEQ     1$              ;NO, CONTINUE

017300  005264  000012                  INC     ERRCNT(R4)      ;STATUS BITS WRONG
017304                                  ERRHRD  60.,MDSER,ERR4
017304  104463                          TRAP    T$ERCODE
017306  000074                          .WORD   60
017310  002753                          .WORD   MDSER
017312  004700                          .WORD   ERR4

017314  010102        1$:       MOV     R1,R2           ;COPY STATUS WORD
017316  042702  177700                  BIC     #177700,R2      ;GET STATE BITS
017322  022702  000034                  CMP     #34,R2          ;COVER CLSD, HEADS OUT, BRUSHES HOME, SEEK TRACK COUNTING?
017326  001444                          BEQ     EXIT            ;YES, EXIT
017330  022702  000035                  CMP     #35,R2          ;COVER CLSD, HEADS OUT, BRUSHES HOME, SEEK LINEAR MODE
017334  001441                          BEQ     EXIT            ;YES, EXIT

017336  005264  000012                  INC     ERRCNT(R4)
017342                                  ERRHRD  70.,MDSER,ERR4
017342  104463                          TRAP    T$ERCODE
017344  000106                          .WORD   70
017346  002753                          .WORD   MDSER
017350  004700                          .WORD   ERR4

017352  000432                          BR      EXIT
```

```
                                .SBTTL  WRITE INTERRUPT SERVICE
                        
017354  042764  000001  000056  AWRITE: BIC     #SKDON,PRFLGS(R4) ;CLEAR SEEK VERIFY FLAG
017362  016401  000042                  MOV     BMP(R4),R1      ;GET WORD COUNT
017366  005401                          NEG     R1              ;MAKE POSITIVE
017370  060164  000006                  ADD     R1,WXFR1(R4)    ;ADD THE BITS
017374  022764  023420  000006          CMP     #10000.,WXFR1(R4)  ;10(5) YET?
017402  101016                          BHI     EXIT            ;NO - EXIT
017404  005264  000010                  INC     WXFR2(R4)       ;YES BUMP 10(10)
017410  162764  023420  000006          SUB     #10000.,WXFR1(R4)  ;10(5) GOES TO ZERO
017416  022764  023420  000010          CMP     #10000.,WXFR2(R4)  ;10(10) YET?
017424  101005                          BHI     EXIT            ;NO - EXIT
017426  005264  000062                  INC     WXFR3(R4)       ;INC 65K (10)(10)
017432  162764  023420  000010          SUB     #10000.,WXFR2(R4)  ;MAKE 10(10)
                        
017440  005764  000036          EXIT:   TST     RETRY(R4)       ;IN PROCESS OF RETRYING?
017444  001414                          BEQ     ERREX           ;NO
017446  026427  000052  004211          CMP     RTYPE(R4),#MTDRV
017454  001406                          BEQ     EXIT1
017456  005264  000014                  INC     SFTCNT(R4)      ;YES, LOG SOFT ERROR
                        
017462                                  ERRSOFT 80.,MSFER,ERR3  ;REPORT SOFT ERROR
017462  104464                          TRAP    T$ERCODE
017464  000120                          .WORD   80
017466  002677                          .WORD   MSFER
017470  004550                          .WORD   ERR3
                        
017472  005064  000036          EXIT1:  CLR     RETRY(R4)       ;CLEAR RETRY
```

```
                                .SBTTL  EXIT FOR INTERRUPT SERVICE
                        
017476  042774  000100  000104  ERREX:  BIC     #INTEN,@DCS(R4)
017504  012600                          MOV     (SP)+,R0
017506  012601                          MOV     (SP)+,R1
017510  012602                          MOV     (SP)+,R2
017512  012603                          MOV     (SP)+,R3
017514  012637  002356                  MOV     (SP)+,INCALL
017520  012637  002354                  MOV     (SP)+,OPCALL
017524  012637  002132                  MOV     (SP)+,WHY
017530  012637  002226                  MOV     (SP)+,TEMP1
017534  012637  002216                  MOV     (SP)+,HDRFND
017540  012637  002220                  MOV     (SP)+,CHKSEC
017544  012637  002310                  MOV     (SP)+,E.MP2
017550  012637  002306                  MOV     (SP)+,E.MP1
017554  012637  002304                  MOV     (SP)+,E.MP
017560  012637  002302                  MOV     (SP)+,E.DA
017564  012637  002300                  MOV     (SP)+,E.BA
017570  012637  002276                  MOV     (SP)+,E.CS
017574  012604                          MOV     (SP)+,R4
017576                                  ENDSRV
017576                          L10023:
017576  000002                          RTI
                        
017600  004537  020600          FINERR: JSR     R5,RCNT         ;CHECK TO SEE IF WE HAVE EXCEEDED
017604  000405                          BR      1$              ;RETRY LIMIT, IF SO 1$ AND REPORT HARD
017606  013764  002276  000116          MOV     E.CS,SOFTCS(R4)
017614  000137  017476                  JMP     ERREX           ;NOT EXCEEDED EXIT
017620  005264  000012          1$:     INC     ERRCNT(R4)      ;INDICATE ERROR
                        
017624                                  ERRHRD  90.,MHDER,ERR13 ;NON-RECOVERABLE ERROR
017624  104463                          TRAP    T$ERCODE
017626  000132                          .WORD   90
017630  003107                          .WORD   MHDER
017632  005312                          .WORD   ERR13
017634  000716                          BR      EXIT1
```

```
                                        .SBTTL  DRIVE ERROR INTERRUPT SERVICE
                                        ;WE HAVE A DRIVE ERROR, LET'S GET THE STATUS
017636  005264  000020          CKDERR: INC     DERCNT(R4)      ;ACCOUNT FOR ERROR
017642  004537  022372                  JSR     R5,GETDST       ;GET DRIVE STATUS
                                        ;REPORT DRIVE ERROR
017646                                  ERRHRD  224.,DRVER,ERR9 ;DRIVE ERROR
017646  104463                          TRAP    T$ERCODE
017650  000340                          .WORD   224
017652  002716                          .WORD   DRVER
017654  005170                          .WORD   ERR9
                                        ;ACT ACCORDINGLY TO DRIVE ERROR
017656  032701  001000                  BIT     #VC,R1          ;VOLUME CHECK?
017662  001027                          BNE     9$              ;YES, GO ISSUE RESET
017664  032701  010000                  BIT     #SKTO,R1        ;SEEK TIME OUT?
017670  001070                          BNE     12$             ;YES, ISSUE RESET
017672  032701  144000                  BIT     #WDE!HCE!SPE,R1 ;WRITE DATA, CURRENT HEAD, SPINDLE?
017676  001130                          BNE     15$             ;GO WAIT FOR HEADS TO UNLOAD
017700  032701  002000                  BIT     #WGE,R1         ;WRITE GATE ERROR
017704  001003                          BNE     20$             ;YES, ISSUE  RESET
017706  004537  022406                  JSR     R5,ISDRST       ;ISSUE RESET
017712  000431                          BR      10$             ;GO CHECK DRIVE READY
017714  004537  022406          20$:    JSR     R5,ISDRST       ;ISSUE RESET
017720  004537  022372                  JSR     R5,GETDST       ;RESET WORK?
017724  032701  002000                  BIT     #WGE,R1         ;WGE CLEAR
017730  001422                          BEQ     10$             ;YES GO CHECK DRIVE READY
017732  012737  003014  002132          MOV     #WGEST,WHY      ;REPORT WGE DIDN'T CLR
017740  000412                          BR      91$             ;DROP DRIVE

017742  004537  022406          9$:     JSR     R5,ISDRST       ;ISSUE RESET
017746  004537  022372                  JSR     R5,GETDST       ;RESET WORK
017752  032701  001000                  BIT     #VC,R1          ;VOL CHK CLEAR
017756  001407                          BEQ     10$             ;YES, CHECK DRIVE READY
017760  012737  002767  002132          MOV     #MVCER,WHY      ;DROP THE DRIVE

017766  004537  021502          91$:    JSR     R5,DRDRV
017772  000137  017472                  JMP     EXIT1
017776  032763  000001  000000  10$:    BIT     #DRDY,CS(R3)    ;DRIVE READY POSTED?
020004  001004                          BNE     101$            ;YES, PRINT RECOVERED

020006  012737  002526  002132          MOV     #DNRDY,WHY
020014  000764                          BR      91$             ;NO, DROP DRIVE

020016                          101$:   PRINTB  #FMT14,#MRDER   ;PRINT DRIVE RECOVERED
020016  012746  003047                  MOV     #MRDER,-(SP)
020022  012746  007132                  MOV     #FMT14,-(SP)
020026  012746  000002                  MOV     #2,-(SP)
020032  010600                          MOV     SP,R0
020034  104014                          EMT     C$PNTB
020036  062706  000006                  ADD     #6,SP
020042  004537  020300                  JSR     R5,GHDR         ;GET THE CURRENT DISK POSITION - HEADER
020046  000137  017600                  JMP     FINERR
020052  012702  000004          12$:    MOV     #4,R2           ;SEEK TIME OUT
020056  004537  022406          13$:    JSR     R5,ISDRST       ;ISSUE DRIVE RESET
```

```
                                                                ;FOUR TIMES BEFORE
020062                                  WAITUS  #15000.         ;DROPPING DRIVE
020062  012700  035230                  MOV     #15000.,R0
020066  104027                          EMT     C$WTU

020070  032763  000001  000000          BIT     #DRDY,CS(R3)    ;DRIVE READY YET?
020076  001006                          BNE     14$             ;YES, CHECK IF ERROR CLEARED
020100  005302                          DEC     R2              ;NO, HAVE WE DONE IT FOUR TIMES
020102  001365                          BNE     13$             ;YET

020104  012737  002725  002132  141$:   MOV     #MDERS,WHY      ;YES, DROP DRIVE
020112  000725                          BR      91$

020114  032763  040000  000000  14$:    BIT     #DERR,CS(R3)    ;DRIVE ERROR SET STILL
020122  001370                          BNE     141$            ;YES, DROP DRIVE
020124                                  PRINTB  #FMT14,#MRDER
020124  012746  003047                  MOV     #MRDER,-(SP)
020130  012746  007132                  MOV     #FMT14,-(SP)
020134  012746  000002                  MOV     #2,-(SP)
020140  010600                          MOV     SP,R0
020142  104014                          EMT     C$PNTB
020144  062706  000006                  ADD     #6,SP
020150  004537  020300                  JSR     R5,GHDR
020154  000137  017440                  JMP     EXIT

020160  012702  000004          15$:    MOV     #4,R2           ;WAIT FOR HEADS TO UNLOAD
020164  004537  022372          16$:    JSR     R5,GETDST       ;GET STATUS
020170  032701  000020                  BIT     #BIT4,R1        ;UNLOAD STATE
020174  001411                          BEQ     17$             ;YES, CONTINUE W/ RECOVERY
020176                                  WAITMS  #15.            ;WAIT A WHILE
020176  012700  000017                  MOV     #15.,R0
020202  104026                          EMT     C$WTM
020204  005302                          DEC     R2              ;WAIT LONG ENOUGH
020206  001366                          BNE     16$             ;NO, GO BACK
020210  012737  003404  002132          MOV     #UNLOAD,WHY     ;DROP DRIVE
020216  000663                          BR      91$

020220  004537  022406          17$:    JSR     R5,ISDRST       ;ISSUE RESET
020224                                  WAITMS  #1.             ;
020224  012700  000001                  MOV     #1.,R0
020230  104026                          EMT     C$WTM
020232  032763  040000  000000          BIT     #DERR,CS(R3)    ;DRIVE ERROR CLEAR?
020240  001321                          BNE     141$            ;NO, DROP DRIVE
020242  012702  000075                  MOV     #61.,R2         ;YES, WAIT 60 SECONDS
020246                          18$:    WAITMS  #10.            ;FOR DRIVE READY TO
020246  012700  000012                  MOV     #10.,R0
020252  104026                          EMT     C$WTM
020254  032763  000001  000000          BIT     #DRDY,CS(R3)    ;COME BACK
020262  001314                          BNE     14$             ;
020264  005302                          DEC     R2
020266  001367                          BNE     18$
020270  012737  003430  002132          MOV     #NOLOAD,WHY     ;NO READY DROP DRIVE
020276  000633                          BR      91$


020300  012763  000210  000000  GHDR:   MOV     #CRDY!RDHDR,CS(R3)
020306  056463  000106  000000          BIS     DRSEL(R4),CS(R3)
```

```
     91 020314  042763  000200  000000          BIC     #200,CS(R3)
     92 020322  004537  022322                  JSR     R5,WTRDY
     93 020326  016301  000006                  MOV     MP(R3),R1
     94 020332  043701  002156                  BIC     SMSK,R1
     95 020336  010164  000124                  MOV     R1,PRPOS(R4)
     96 020342  012764  004211  000052          MOV     #MTDRV,RTYPE(R4)  ;SETUP DRIVE ERROR
     97 020350  000205                          RTS     R5
```

```
                                .SBTTL  BUFFER GENERATION ROUTINE FOR THE 'WRITE' FUNCTION
020352                          STARS
                                ;;*****************************************************************
                                ;WRBUF -- ROUTINE TO WRITE A BUFFER INTO MEMORY.  USES WORD COUNT AND BUS
                                ;       ADDRESS FROM DRIVE BUFFER (R4).  WILL WRITE RANDOM FROM ONE OF
                                ;       8 PATTERNS.  USED BY WRITE FUNCTION AND WRPACK ROUTINE.
020352                          STARS
                                ;;*****************************************************************

020352  005737  002172        WRBUF:  TST     REGEN           ;REBUILD THE DATA BUFFER?
020356  001507                        BEQ     9$              ;NO --EXIT
020360  010346                        MOV     R3,-(SP)        ;SAVE REGISTERS
020362  010246                        MOV     R2,-(SP)
020364  010146                        MOV     R1,-(SP)
020366  010046                        MOV     R0,-(SP)
020370  016402  000042                MOV     BMP(R4),R2      ;R2 HAS TOTAL WORDS TO SET UP FOR
020374  005402                        NEG     R2              ;POSITIVE NUMBER
020376  017401  00011C                MOV     @BBA(R4),R1     ;WHERE BUFFER IS
020402  020227  000200          2$:   CMP     R2,#128.        ;MORE THAN 128 WORDS
020406  002015                        BGE     4$              ;YES, BRANCH
020410  020227  000003                CMP     R2,#3           ;GREATER THAN THREE WORDS
020414  002005                        BGE     3$              ;YES, BRANCH
020416  062702  000003                ADD     #3,R2           ;ADD 3
020422  162764  000003  000042        SUB     #3,BMP(R4)      ;WC UP BY 3
020430  010221                  3$:   MOV     R2,(R1)+        ;STORE WC
020432  005302                        DEC     R2              ;ACCOUNT FOR WC
020434  010237  002240                MOV     R2,TEMP6        ;LOAD DOWN COUNTER
020440  000405                        BR      5$
020442  012737  000177  002240  4$:   MOV     #127.,TEMP6     ;LOAD DOWN COUNTER
020450  012721  000200                MOV     #128.,(R1)+
020454  005737  010324          5$:   TST     T.RAN           ;RANDOM SELECT OF PATTERNS
020460  001003                        BNE     55$             ;YEA
020462  013703  010326                MOV     T.PAT,R3        ;NO GET PATTERN OPERATOR
020466  000406                        BR      56$             ;WANTS TO USE
020470  004537  022464          55$:  JSR     R5,RAND         ;GET RANDOM # FOR PATTERN
020474  013703  002146                MOV     LONUM,R3        ;GET RANDOM PATTERN
020500  042703  177770                BIC     #177770,R3      ;0,7
020504  006303                  56$:  ASL     R3              ;WORD OFFSET
020506  062703  026066                ADD     #PATLST,R3      ;GET PATTERN LIST
020512  011303                        MOV     (R3),R3         ;GET LIST ADDRESS
020514  010337  002242                MOV     R3,TEMP7        ;STOR FOR RECALL
020520  010321                        MOV     R3,(R1)+        ;LOAD IT
020522  005337  002240                DEC     TEMP6           ;ACCOUNT FOR IT
020526  013703  002242          6$:   MOV     TEMP7,R3        ;PATTERN START
020532  012737  000020  002244        MOV     #16.,TEMP8      ;16 ENTRIES
020540  012321                  7$:   MOV     (R3)+,(R1)+     ;STORE PATTERN
020542  005337  002240                DEC     TEMP6           ;DOWN COUNT
020546  001404                        BEQ     8$              ;DONE?
020550  005337  002244                DEC     TEMP8           ;DONE WITH PATTERN
020554  001371                        BNE     7$              ;NO, GO BACK
020556  000763                        BR      6$              ;RESTART PATTERN
020560  162702  000200          8$:   SUB     #128.,R2        ;ANOTHER SECTOR TO USE
020564  003306                        BGT     2$              ;YES GO BACK
020566  012600                        MOV     (SP)+,R0        ;RESTORE REGISTERS
020570  012601                        MOV     (SP)+,R1
020572  012602                        MOV     (SP)+,R2
020574  012603                        MOV     (SP)+,R3
```

```
     56 020576  000205                    9$:     RTS     R5
     57
     58                                    .SBTTL  RETRY LIMIT ROUTINE
     59
     60                                    ;RETRY BUMP, TWO RETURNS - CALL +2 - RETRY EXCEEDED
     61                                    ;                          CALL +4 - CONTINUE RETRY
     62
     63 020600  026437  000036  010246    RCNT:   CMP     RETRY(R4),LIMIT ;LIMIT REACHED?
     64 020606  001403                            BEQ     1$              ;YES TAKE FIRST RETURN
     65 020610  005264  000036                    INC     RETRY(R4)       ;ACCOUNT FOR RETRY
     66 020614  005725                            TST     (R5)+           ;NEXT RETURN
     67 020616  000205                    1$:     RTS     R5              ;RETURN
     68
     69                                    .SBTTL  LIST OF FUNCTION ROUTINES
     70
     71                                    ;WE GO THRU THIS LIST WHEN CALLED IN "GETFNC"
     72                                    ;LIST IS IN NUMERICAL ORDER 1-6
     73
     74 020620  000000                    LIST:   .WORD   0
     75 020622  014036                            SKWRT           ;SEEK - WRITE DATA - WRITE CHECK
     76 020624  014076                            SKRD            ;SEEK - READ DATA
     77 020626  014246                            SKRH            ;SEEK - READ HDR - READ W/NO HDR CMP - GET STATUS
     78 020630  014036                            SKWRT           ;SEEK - WRITE DATA - WRITE CHECK
     79 020632  014076                            SKRD            ;SEEK - READ DATA
     80 020634  014126                            SKRDRD          ;SEEK - READ DATA - READ DATA
```

```
                                .SBTTL  BAD SECTOR FILE ROUTINE
020636                          STARS
                                ;;*****************************************************************
                                ;RDBDSC -- ROUTINE TO RECOVER BAD SECTOR FILE AND SAVE IT FOR
                                ;COMPARISON UPON ERROR ON READS/WRITES & FOR THE SEEK FUNCTION. WE
                                ;WILL ONLY RESERVE SPACE FOR 16 BAD SECTORS PER DRIVE AND 1 ENTRY FOR
                                ;THE BAD SECTOR FILE AREA POINTER - LAST TRACK ON THE CARTRIDGE.
                                ;WE WILL ISSUE A DRIVE RESET FIRST, READ HEADER, POSITION TO THE LAST
                                ;TRACK (CYLINDER 255. OR 511., 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               ;GET THE BAD SECT FILE ENTRYS
                                ;
                                ;THE BAD SECTOR FILE (BOTH FACTORY AND FIELD) LOOKS LIKE THIS:
                                ;
                                ;       SERIAL NUMBER LOW 5 DIGITS (OCTAL SERIAL NUMBER)
                                ;       SERIAL NUMBER HIGH 5 DIGITS
                                ;
                                ;       0'S
                                ;       0'S
                                ;
                                ;       ENTRY - CYLINDER # FROM 0 TO 1777 MAX (RL02) OR 777 (RL01)
                                ;       ENTRY - HEAD & SECTOR NUMBER
                                ;
                                ;       ENTRY - CYL
                                ;       ENTRY - HEAD & SECTOR
                                ;
                                ;       -1 ...END OF ENTRYS
                                ;       -1 ...TO WORD 256. (END OF SECOND SECTOR IN PAIR)
020636                          STARS
                                ;;*****************************************************************

020636  010046          RDBDSC: MOV     R0,-(SP)        ;SAVE REGISTERS
020640  010146                  MOV     R1,-(SP)        ;
020642  010246                  MOV     R2,-(SP)
020644  010346                  MOV     R3,-(SP)
020646  004537  022406  21$:     JSR     R5,ISDRST       ;ISSUE A DRIVE RESET
020652  012764  000010  000044          MOV     #RDHDR,FUNC (R4);READ HEADER TO FIND POSITION
020660  004537  015226          JSR     R5,LDFUNC       ;ON  DISK
020664  004537  022322          JSR     R5,WTRDY

020670  016300  000006          MOV     MP(R3),R0       ;GET HEADER AND CALCULATE
020674  022764  000001  000120  CMP     #1,TDR(R4)      ;RL02 TYPE DRIVE?
020702  001005                  BNE     23$             ;JUMP IF RL02
020704  043700  002150          BIC     CYLMSK,R0       ;HERE FOR RL01
020710  012701  077600          MOV     #77600,R1
020714  000404                  BR      25$
020716  043700  002154  23$:     BIC     CMSK,R0         ;HERE FOR RL02
020722  012701  177600          MOV     #177600,R1
020726  160001          25$:     SUB     R0,R1
020730  010164  000040          MOV     R1,BDA(R4)
020734  052764  000025  000040  BIS     #SKHS!SIGN!MK,BDA(R4)
020742  012764  000006  000044  MOV     #SEEK,FUNC(R4)
020750  004537  015226          JSR     R5,LDFUNC       ;SEEK TO THE BAD SECTOR FILE AREA
020754  004537  022322          JSR     R5,WTRDY        ;WAIT FOR DRIVE READY
020760  012764  000010  000044  MOV     #RDHDR,FUNC(R4)
```

```
                                                JSR     R5,LDFUNC       ;READ A HEADER ON THE BSF
                                                JSR     R5,WTRDY        ;WAIT FOR DRIVE READY
                                                MOV     MP(R3),R0       ;GET THE HEADER WORD READ
                                                BIC     #77,R0          ;CLEAR SECTOR NUMBER READ
                                                CMP     #1,TDR(R4)      ;DRIVE = RL01?
                                                BNE     300$            ;NO - MUST BE AN RL02
                                                CMP     #77700,R0       ;YES - ON BSF AREA?
                                                BNE     21$             ;NO - SEEK AGAIN
                                                MOV     #77700,BDA(R4)  ;SAVE THIS HEADER FOR READ COMMAND
                                                BR      555$
                                        300$:   CMP     #177700,R0      ;RL02 BSF AREA?
                                                BNE     21$             ;NO - SEEK AGAIN
                                                MOV     #177700,BDA(R4) ;YES - SAVE FOR THE READ COMMAND
                                        555$:   MOV     #-256.,BMP(R4)  ;SETUP FOR A 2 SECTOR READ IN BSF
                                                MOV     #READ,FUNC(R4)  ;GET THE READ FUNCTION #

                                                CLR     TEMP3           ;MANUFACTURING/FIELD FILE SWITCH
                                                MOV     #HWSEC,WHY      ;START WITH MANUFACTURING BAD
                                                MOV     BSECPT(R4),R2   ;INITIALIZE LIST TO ALL 1'S
                                                MOV     #17.,R0         ;SIXTEEN ENTRIES + 1 FOR BSF POINTER
                                        11$:    MOV     #-1,(R2)+       ;INIT STORAGE TO -1'S
                                                DEC     R0              ;DONE?
                                                BNE     11$             ;NO - DO THE NEXT ONE

                                                MOV     BSECPT(R4),R2   ;GET POINTER TO LIST TO STORE BSF ENTRYS
                                                MOV     BDA(R4),(R2)+   ;SAVE 1ST ENTRY AS BSF POINTER
                                                MOV     #16.,R0         ;SIXTEEN ENTRIES
                                        4$:     JSR     R5,LDFUNC       ;READ THE BSF SECTOR PAIR
                                                JSR     R5,WTRDY        ;WAIT FOR DRIVE READY

                                                TST     @DCS(R4)        ;WAS THE READ GOOD?
                                                BPL     3$              ;YES

                                                JSR     R5,ISDRST       ;NO - ISSUE A DRIVE RESET
                                                ADD     #4,BDA(R4)      ;POINT TO NEXT SECTOR
                                                TST     TEMP3           ;MANUFACTURING OR FIELD BAD
                                                BEQ     5$              ;MANUFACTURING = 0
                                                MOV     #SWSEC,WHY      ;FIELD BAD
                                                CMP     #1,TDR(R4)      ;DRIVE = RL01?
                                                BNE     400$            ;NO - MUST BE RL02
                                                CMP     #77750,BDA(R4)  ;YES - AT END OF FIELD FILE?
                                                BNE     4$              ;NO - CONTINUE
                                                BR      6$              ;DROP DRIVE AND EXIT

                                        5$:     CMP     BDA(R4),#77724  ;AT END OF MANUFACTURING BAD
                                                BR      55$             ;SEE IF DONE
                                        400$:   CMP     #177750,BDA(R4) ;AT END OF FIELD BAD FOR RL02
                                                BNE     4$              ;NO GO BACK FOR NEXT
                                                BR      6$              ;DROP THE DRIVE AND EXIT
                                                CMP     BDA(R4),#177724 ;AT END OF MANUFACTURING BAD?
                                        55$:    BNE     4$              ;BR IF NOT DONE
                                                BR      6$              ;YES - REPORT ERROR AND EXIT

                                        3$:     MOV     @BBA(R4),R1     ;START OF BSF ENTRY LIST
                                                MOV     (R1)+,SERNM1(R4)  ;GET LOW PART OF SERIAL #
                                                MOV     (R1)+,SERNM2(R4)  ;GET HIGH PART OF SERIAL #
                                                CMP     (R1)+,(R1)+     ;SKIP PAST JUNK
```

```
021272  012137  002226          1$:     MOV     (R1)+,TEMP1        ;GET CYLINDER
021276  100444                          BMI     2$                 ;END OF THE ENTRYS?
021300  012137  002230                  MOV     (R1)+,TEMP2        ;NO - GET HEAD (0 OR 1) & SECTOR NUMBER
021304  000337  002226                  SWAB    TEMP1              ;PUT CYLINDER IN HIGH BYTE
021310  000241                          CLC
021312  006037  002226                  ROR     TEMP1
021316  103003                          BCC     111$
021320  052737  100000  002226          BIS     #BIT15,TEMP1
021326  013712  002226          111$:   MOV     TEMP1,(R2)         ;STORE THE CYLINDER PART
021332  013737  002230  002226          MOV     TEMP2,TEMP1        ;GET SECTOR
021340  042737  177700  002226          BIC     #177700,TEMP1      ;LEAVE ONLY SECTOR
021346  053712  002226                  BIS     TEMP1,(R2)         ;SET IN SECTOR BITS
021352  006237  002230                  ASR     TEMP2
021356  006237  002230                  ASR     TEMP2              ;POSITION THE HEAD SELECT BIT
021362  042737  177677  002230          BIC     #177677,TEMP2      ;CLEAR ALL OTHER BITS
021370  053722  002230                  BIS     TEMP2,(R2)+        ;SET IN HEAD
021374  005300                          DEC     R0                 ;COUNT THIS ENTRY FROM BSF
021376  001335                          BNE     1$                 ;ALLOW MORE ENTRYS?
021400  012737  003525  002132          MOV     #MBDMSC,WHY        ;MORE THAN 16 BAD SECTORS
021406  000422                          BR      6$                 ;DROP THE DRIVE & ERROR EXIT

021410  005737  002232          2$:     TST     TEMP3              ;SWITCH TO FIELD BAD OR QUIT
021414  001021                          BNE     7$                 ;QUIT, 7$
021416  022764  000001  000120          CMP     #1,TDR(R4)         ;DRIVE = RL01?
021424  001004                          BNE     350$               ;NO - MUST BE AN RL02
021426  012764  077724  000040          MOV     #77724,BDA(R4)     ;YES - POINT TO FIELD BSF 1ST SECTOR
021434  000403                          BR      36$
021436  012764  177724  000040  350$:   MOV     #177724,BDA(R4)    ;POINT TO 1ST SECT IN FIELD FILE FOR RL02
021444  012737  000001  002232  36$:    MOV     #1,TEMP3           ;INDICATE NOW DOING FIELD BSF
021452  000627                          BR      4$                 ;PROCESS THE FIELD BSF

                                ;HERE TO DROP THE DRIVE IF MORE THAN 16. ENTRYS OR IF CAN'T FIND A BSF

021454  004537  021502          6$:     JSR     R5,DRDRV           ;DROP THE DRIVE

                                ;HERE TO PUT HEADS 'HOME' AND TO EXIT

021460  004537  023752          7$:     JSR     R5,HDHOME          ;BRINGS HEADS HOME
021464  012603                          MOV     (SP)+,R3
021466  012602                          MOV     (SP)+,R2
021470  012601                          MOV     (SP)+,R1
021472  012600                          MOV     (SP)+,R0
021474  000205                          RTS     R5
```

```
                                .SBTTL  ROUTINE TO DROP DRIVE
021476                          STARS
                                ;;*****************************************************************
                                ;DRDRV -- ROUTINE TO DROP A DRIVE FROM RUNNING
                                ;       R4 HAS BUFFER POINTER OF DRIVE TO DROP
                                ;       WE CLEAR BIT IN "DRUT", NOT "DRPRS"
021476                          STARS
                                ;;*****************************************************************

021476  005237  002354          ODRDRV: INC     OPCALL
021502  010146                  DRDRV:  MOV     R1,-(SP)
021504  010246                          MOV     R2,-(SP)        ;SAVE REGISTERS
021506  010346                          MOV     R3,-(SP)
021510  005237  002356                  INC     INCALL
021514  005003                          CLR     R3
021516  012702  026514                  MOV     #DRBUF,R2       ;START OF DRIVE BUFFERS
021522  012701  000001                  MOV     #1,R1           ;MASK
021526  020402                  1$:     CMP     R4,R2           ;IS THIS THE DRIVE?
021530  001405                          BEQ     2$              ;YES GO DROP IT
021532  005203                          INC     R3
021534  006301                          ASL     R1              ;NO SHIFT MASK
021536  062702  000126                  ADD     #PRPOS+2,R2     ;NEXT BUFFER
021542  000771                          BR      1$              ;GO BACK

021544  005737  002354          2$:     TST     OPCALL          ;CALLED VIA OPERATOR?
021550  001002                          BNE     6$              ;YES - SKIP CODE
021552                                  DODU    R3              ;NO - CALLED BY DIAGNOSTIC
021552  010300                          MOV     R3,R0
021554  104053                          EMT     C$DODU
021556  005037  002356          6$:     CLR     INCALL
021562  005037  002354                  CLR     OPCALL
021566  113764  002274  000070          MOVB    HOUR,DPHOUR(R4) ;TIME AT WHICH IT WAS DROPPED
021574  113764  002272  000071          MOVB    MINUTE,DPMIN(R4)  ;HOUR/MINUTE
021602  001002                          BNE     3$              ;IF MINUTE 0,
021604  105264  000071                  INCB    DPMIN(R4)       ;MAKE 1.
021610  140137  002136          3$:     BICB    R1,DRUT         ;CLEAR THE DRIVE FROM BIT MAP
021614                                  PRINTF  #FMT14C         ;PRINT A <CR> & <LF>
021614  012746  007150                  MOV     #FMT14C,-(SP)
021620  012746  000001                  MOV     #1,-(SP)
021624  010600                          MOV     SP,R0
021626  104017                          EMT     C$PNTF
021630  062706  000004                  ADD     #4,SP
021634  004737  005606                  JSR     PC,LINE2
021640                                  PRINTF  #FMT7,#DROP,WHY
021640  013746  002132                  MOV     WHY,-(SP)
021644  012746  004134                  MOV     #DROP,-(SP)
021650  012746  006567                  MOV     #FMT7,-(SP)
021654  012746  000003                  MOV     #3,-(SP)
021660  010600                          MOV     SP,R0
021662  104017                          EMT     C$PNTF
021664  062706  000010                  ADD     #10,SP
021670                                  PRINTF  #FMTS1
021670  012746  007401                  MOV     #FMTS1,-(SP)
021674  012746  000001                  MOV     #1,-(SP)
021700  010600                          MOV     SP,R0
021702  104017                          EMT     C$PNTF
021704  062706  000004                  ADD     #4,SP
```

```
     37
     38 021710  004737  012442                JSR     PC,REPORT
     39
     40 021714  012603                        MOV     (SP)+,R3
     41 021716  012602                        MOV     (SP)+,R2        ;RESTORE REGISTERS
     42 021720  012601                        MOV     (SP)+,R1
     43
     44 021722  000205                        RTS     R5
```

```
                                .SBTTL  ROUTINE TO CHECK DATA

                                ;ROUTINE TO CHECK DATA ON READ

021724  005037  002172      CKDATA: CLR     REGEN           ;CLEAR THE REGENERATE DATA FLAG
021730  005737  010262              TST     CMRD            ;DO WE WANT TO CHECK ANY?
021734  001001                  BNE     10$             ;YES - SEE IF FORCED EXIT
021736  000205                  RTS     R5              ;NO - EXIT NOW
021740  005737  002174      10$:    TST     KILLDC          ;INHIBIT FLAG SET?
021744  001401                  BEQ     97$             ;NOPE - OK TO PROCEED
021746  000205                  RTS     R5              ;NO, EXIT

021750                  97$:    SETPRI  #340
021750  012700  000340              MOV     #340,R0
021754  104041                  EMT     C$SPRI
021756  017402  000110              MOV     @BBA(R4),R2     ;BUFFER START
021762  016437  000042  002226      MOV     BMP(R4),TEMP1   ;WORDS READ IN
021770  005437  002226              NEG     TEMP1           ;MAKE POSITIVE
021774  013737  010264  002230      MOV     DELMT,TEMP2     ;# ERRORS TO BE PRINTED
022002  005037  002222              CLR     DECNT           ;INIT ERROR COUNT
022006  013737  010262  002232      MOV     CMRD,TEMP3      ;# WORDS TO BE COMPARED
022014  012737  000176  002224  96$:    MOV     #126.,TEMP0     ;126 WORDS
022022  012201                  MOV     (R2)+,R1        ;NON-ZERO WORDS
022024  005337  002226              DEC     TEMP1
022030  001522                  BEQ     CEND
022032  005301                  DEC     R1
022034  012237  002234              MOV     (R2)+,TEMP4     ;PATTERN ADDRESS

                                ;MAKE SURE PATTERN ADDRESS IS LEGAL

022040  012700  026066              MOV     #PATLST,R0      ;GET LIST OF PATTERNS
022044  012703  000010              MOV     #8.,R3          ;ONLY EIGHT
022050  022037  002234      98$:    CMP     (R0)+,TEMP4     ;FOUND IT YET
022054  001414                  BEQ     99$             ;YES, CONTINUE
022056  005303                  DEC     R3              ;NO, EXHAUST LIST YET
022060  001373                  BNE     98$             ;NO, GO BACK

022062  005237  002172              INC     REGEN           ;SET THE DATA REGENERATE FLAG
022066  024242                  CMP     -(R2),-(R2)
022070                          ERRHRD  180.,NOREV,ERR12
022070  104463                  TRAP    T$ERCODE
022072  000264                  .WORD   180
022074  003466                  .WORD   NOREV
022076  005304                  .WORD   ERR12
022100  004537  025070              JSR     R5,STDMP
022104  000205                  RTS     R5

022106  005301          99$:    DEC     R1              ;ACCOUNT FOR PATTERN ADDRESS
022110  013703  002234              MOV     TEMP4,R3        ;GET ADDRESS
022114  005337  002226              DEC     TEMP1           ;ACCOUNT ONCE AGAIN
022120  012737  000020  002236      MOV     #16.,TEMP5      ;16 ENTRIES TO PATTERN
022126  005737  002226      1$:     TST     TEMP1           ;ANY WORDS READIN LEFT?
022132  001461                  BEQ     CEND            ;NO, GO TO END
022134  005737  002232              TST     TEMP3           ;HAVE WE EXHAUSTED COMPARE LIMIT?
022140  001456                  BEQ     CEND            ;YES GO TO END
022142  005701                  TST     R1              ;WE CHECKING PATTERN OR ZERO FILL?
022144  001416                  BEQ     3$              ;ZERO FILL SKIP
```

```
                                                DEC     R1              ;PATTERN
                                                TST     TEMP5           ;WITHIN PATTERN
                                                BNE     2$              ;YES SKIP
                                                MOV     TEMP4,R3        ;NO, START OVER
                                                MOV     #16.,TEMP5      ;16 ENTRIES
                                        2$:     MOV     (R3)+,GDDAT     ;GET PATTERN
                                                DEC     TEMP5           ;DOWN COUNT
                                                BR      4$
                                        3$:     CLR     GDDAT           ;ZERO FILL
                                        4$:     CMP     GDDAT,(R2)      ;CORRECT DATA
                                                BEQ     5$              ;YES YES NEXT
                                                INC     REGEN           ;NO - SET REGENERATE FLAG FOR WRT OPERATION
                                                INC     DECNT           ;COUNT THE DATA ERROR
                                                INC     DATCER(R4)      ;COUNT ERROR FOR THIS DRIVE
                                                TST     TEMP2           ;DO WE WANT TO PRINT IT
                                                BEQ     5$              ;NO,SKIP

                                                ERRHRD  185.,MDCER,ERR8
                                                TRAP    T$ERCODE
                                                .WORD   185
                                                .WORD   MDCER
                                                .WORD   ERR8
                                                DEC     TEMP2           ;ACCOUNT FOR PRINT

                                        5$:     DEC     TEMP1           ;WORDS READ IN
                                                BEQ     CEND
                                                TST     (R2)+           ;NEXT WORD
                                                DEC     TEMP0
                                                BEQ     96$
                                                DEC     TEMP3           ;WORDS TO CHECK
                                                BR      1$

                                        CEND:   TST     DECNT           ;DO WE WANT TO PRINT SUMMARY
                                                BEQ     1$              ;NO,EXIT
                                                NEG     BMP(R4)         ;MAKE POSITIVE WORD COUNT
                                                ERRHRD  190.,MDCER,ERR6 ;DATA ERROR SUMMARY
                                                TRAP    T$ERCODE
                                                .WORD   190
                                                .WORD   MDCER
                                                .WORD   ERR6

                                        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

022322  010046          WTRDY:  MOV     R0,-(SP)        ;SAVE REGISTERS
022324  010146                  MOV     R1,-(SP)
022326  012701  001750          MOV     #1000.,R1       ;WAIT A WHILE
022332                  1$:     WAITUS  #2.
022332  012700  000002          MOV     #2.,R0
022336  104027                  EMT     C$WTU
022340  032774  000200  000104  BIT     #CRDY,@DCS(R4)  ;READY SET?
022346  001006                  BNE     2$              ;YES, EXIT
022350  005301                  DEC     R1              ;TIMED OUT?
022352  001367                  BNE     1$              ;NO GO BACK

022354                          ERRDF   1002.,NOCRDY,ERR12
022354  104462                  TRAP    T$ERCODE
022356  001752                  .WORD   1002
022360  002516                  .WORD   NOCRDY
022362  005304                  .WORD   ERR12

022364  012601          2$:     MOV     (SP)+,R1        ;RESTORE REGISTERS
022366  012600                  MOV     (SP)+,R0
022370  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

022372  016403  000104          GETDST: MOV     DCS(R4),R3
022376  012763  000003  000004          MOV     #GSBIT,DA(R3)
022404  000405                  ISDRST: BR      CSTUFF
022406  016403  000104          ISDRST: MOV     DCS(R4),R3
022412  012763  000013  000004          MOV     #DRST,DA(R3)
022420  012763  000204  000000  CSTUFF: MOV     #CRDY!GSTAT,CS(R3)
022426  056463  000106  000000          BIS     DRSEL(R4),CS(R3)
022434  042763  000200  000000          BIC     #CRDY,CS(R3)
022442  004537  022322                  JSR     R5,WTRDY
022446  022763  000013  000004          CMP     #DRST,DA(R3)
022454  001402                          BEQ     1$
022456  016301  000006                  MOV     MP(R3),R1
022462  000205                  1$:     RTS     R5

022464                          STARS
                                ;;*****************************************************************
                                ;RAND --  ROUTINE TO GENERATE A RANDOM NUMBER
022464                          STARS
                                ;;*****************************************************************

022464  010146                  RAND:   MOV     R1,-(SP)
022466  010246                          MOV     R2,-(SP)
022470  010346                          MOV     R3,-(SP)

022472  013703  002146                  MOV     LONUM,R3
022476  013701  002144                  MOV     HINUM,R1
022502  012702  177771                  MOV     #-7,R2
022506  006303                  1$:     ASL     R3
022510  006101                          ROL     R1
022512  005202                          INC     R2
022514  001374                          BNE     1$
022516  063703  002146                  ADD     LONUM,R3
022522  005501                          ADC     R1
022524  063701  002144                  ADD     HINUM,R1
022530  062703  001057                  ADD     #1057,R3
022534  005501                          ADC     R1
022536  062701  047401                  ADD     #47401,R1
022542  010337  002144                  MOV     R3,HINUM
022546  010137  002146                  MOV     R1,LONUM
022552  012603                          MOV     (SP)+,R3
022554  012602                          MOV     (SP)+,R2
022556  012601                          MOV     (SP)+,R1
022560  000205                          RTS     R5
```

```
                                .SBTTL  ROUTINE TO WRITE PACKS INITIALLY
022562                          STARS
                                ;;*****************************************************************
                                ;WRPACK -- ROUTINE TO WRITE PACK WITH PATTERN, ALL TRACKS WILL BE
                                ;         WRITTEN (EXCEPT BAD SECTOR TRACK)
                                ;         FORMAT IS # OF WORDS (WORD 1), PATTERN ADDRESS (WORD 2)
                                ;         PATTERN (WORDS 3 - 128)
                                ;         WE WILL ATTEMPT TO WRITE MULTIPLE SECTORS AT A TIME
                                ;         (MINIMUM 10 SECTORS) IF AN ERROR OCCURS WE WILL THEN
                                ;         WRITE INDIVIDUAL SECTORS FOR THAT TRACK.  WE DO WRITES,
                                ;         READS AND INCORE COMPARISONS TO VERIFY.
                                ;
                                ;         CALL:   JSR     R5,WRPACK       ;WRITE THE PACK SELECTED
022562                          STARS
                                ;;*****************************************************************

022562  010046          WRPACK: MOV     R0,-(SP)        ;SAVE REGISTERS
022564  010146                  MOV     R1,-(SP)
022566  010246                  MOV     R2,-(SP)
022570  010346                  MOV     R3,-(SP)
022572  016446  000110          MOV     BBA(R4),-(SP)
022576  005764  000122          TST     WRIPG(R4)       ;SEE IF WRITE IN PROGRESS
022602  001016                  BNE     1$              ;JUMP IF DON'T WANT MESSAGE ON RECOVERY
022604                          PRINTF  #FMT18,#MSWRPK
022604  012746  004236          MOV     #MSWRPK,-(SP)
022610  012746  007335          MOV     #FMT18,-(SP)
022614  012746  000002          MOV     #2,-(SP)
022620  010600                  MOV     SP,R0
022622  104017                  EMT     C$PNTF
022624  062706  000006          ADD     #6,SP
022630  004537  023630          JSR     R5,GETSYS       ;GET THE CURRENT RUN TIME
022634  004737  005606          JSR     PC,LINE2        ;PRINT TIME-RCLS & DRIVE ID
022640  004537  023752  1$:     JSR     R5,HDHOME       ;HEADS HOME

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

022644  005037  002226          CLR     TEMP1           ;TEMP1=HEAD
022650  005001                  CLR     R1              ;R1=CYL
022652  022764  000001  000120  CONWR:  CMP     #1,TDR(R4)
022660  001007                  BNE     45$
022662  022701  077600          CMP     #077600,R1
022666  001023                  BNE     STWRT
022670  005737  002226          TST     TEMP1
022674  001420                  BEQ     STWRT
022676  000406                  BR      ENDWR
022700  022701  177600  45$:    CMP     #177600,R1
022704  001014                  BNE     STWRT           ;NO GO WRITE TRACK
022706  005737  002226          TST     TEMP1           ;YES, CHECK IF HEAD = 1?
022712  001411                  BEQ     STWRT           ;HEAD = 0 GO WRITE
022714  004537  023752  ENDWR:  JSR     R5,HDHOME       ;HEADS HOME
022720  012664  000110          MOV     (SP)+,BBA(R4)
022724  012603                  MOV     (SP)+,R3
022726  012602                  MOV     (SP)+,R2
022730  012601                  MOV     (SP)+,R1
```

```
                                        MOV     (SP)+,R0
                                        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.

                                STWRT:  CLR     R2              ;INITIAL SECTOR 0
                                        MOV     #BUF1,BBA(R4)   ;BUFFER START
                                        MOV     #-1280.,BMP(R4) ;10 SECTORS
                                        INC     REGEN           ;SET THE GENERATE BUFFER FLAG
                                        JSR     R5,WRBUF        ;WRITE BUFFER INTO MEMORY
                                201$:   MOV     R1,BDA(R4)      ;SET UP SECTOR
                                        BIS     TEMP1,BDA(R4)
                                        TST     WRIPG(R4)       ;WRITE IN PROGRESS?
                                        BEQ     762$            ;NO - JUMP OVER
                                        CMP     PRPOS(R4),BDA(R4) ;YUP - ON CYLINDER NOW?
                                        BEQ     762$            ;YUP - WRITE THIS AREA
                                        JMP     952$            ;NO - LOOK AT NEXT AREA ON DRIVE
                                762$:   BIS     R2,BDA(R4)
                                        MOV     #BUF1,BBA(R4)   ;SET UP TO WRITE
                                        MOV     #WRITE,FUNC(R4) ;WRITE
                                        JSR     R5,LDFUNC
                                        JSR     R5,WTRDY        ;WAIT FOR READY
                                        TST     @DCS(R4)        ;ERROR
                                        BPL     203$
                                205$:   JSR     R5,ISDRST
                                        BR      2$

                                203$:   MOV     #WRCHK,FUNC(R4)
                                        JSR     R5,LDFUNC
                                        JSR     R5,WTRDY
                                        TST     @DCS(R4)        ;ERROR
                                        BMI     205$            ;YES GO DO SECTORS INDIVIDUALLY


                                        ADD     #10.,R2         ;NEXT GROUP
                                        CMP     #40.,R2         ;DONE?
                                        BNE     201$            ;NO, GO BACK
                                        JMP     952$            ;YES NEXT TRACK

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

                                2$:     CLR     R2              ;R2 = SECTOR

                                        MOV     #-128.,BMP(R4)  ;LOAD WORD COUNT
                                3$:     MOV     R1,BDA(R4)      ;SETUP DISK ADDRESS
                                        BIS     TEMP1,BDA(R4)
                                        BIS     R2,BDA(R4)

                                        MOV     #BUF1,BBA(R4)
                                        JSR     R5,WRBUF        ;WRITE A BUFFER
                                91$:    CLR     RWCNT           ;CLEAR RETRYS OUT
                                98$:    CLR     DECNT           ;
                                96$:    MOV     #WRITE,FUNC(R4) ;WRITE FUNCTION
                                        JSR     R5,LDFUNC
                                        JSR     R5,WTRDY        ;WAIT FOR WRITE TO FINISH
```

```
                                                TST     @DCS(R4)          ;ERROR ON WRITE?
                                                BPL     85$               ;NO, GO READ

                                                MOV     BDA(R4),CHKSEC    ;YES, CHECK IF SECTOR IS IN
                                                JSR     R5,CKBDSC         ;BAD SECTOR FILE
                                                TST     HDRFND            ;IF SET, IT WAS
                                                BNE     802$              ;YES GO TO NEXT SECTOR

                                                INC     DECNT             ;NO, GIVE IT 3 TRYS TOTAL
                                                CMP     DECNT,#3.         ;IT MAY HAVE BEEN NOISE.
                                                BEQ     801$              ;BR IF AT RETRY LIMIT - BAD SECTOR
                                                JSR     R5,ISDRST         ;RESET THE DRIVE & TRY AGAIN
                                                BR      96$               ;TRY RECOVERY AGAIN

                                        85$:    CLR     RECNT             ;CLEAR RETRY COUNT
                                        80$:    MOV     #WRCHK,FUNC(R4)   ;READ/VERIFY THE 1 SECTOR WRITTEN
                                                JSR     R5,LDFUNC         ;ISSUE A WRITE-CHECK FUNCTION
                                                JSR     R5,WTRDY          ;WAIT FOR DRIVE READY

                                                TST     @DCS(R4)          ;ERROR ON READ?
                                                BPL     95$               ;BR IF OK ... GET THE NEXT SECTOR

                                                MOV     BDA(R4),CHKSEC    ;CHECK IF SECTOR IS
                                                JSR     R5,CKBDSC         ;A KNOWN BAD SECTOR
                                                TST     HDRFND            ;IT WAS THEN
                                                BNE     802$              ;GO TO NEXT SECTOR

                                                INC     RECNT             ;GIVE IT ANOTHER CHANCE
                                                CMP     RECNT,#16.        ;16 RE-READS BEFORE HARD ERROR
                                                BEQ     801$              ;REPORT ERROR IF AT RETRY LIMIT
                                                JSR     R5,ISDRST         ;RESET THE DRIVE
                                                BR      80$               ;AND RETRY AGAIN

                                        801$:   JSR     R5,INBAD          ;REPORT THE BAD SECTOR
                                        802$:   JSR     R5,ISDRST         ;RESET THE DRIVE FOR THE NEXT OPERATION

                                        95$:    ADD     #10.,R2           ;NEXT SECTOR (OFFSET BY 10)
                                                CMP     R2,#39.           ;DONE WITH TRACK?
                                                BGT     951$              ;YES NEXT TRACK
                                                JMP     3$                ;NO GO BACK FOR NEXT SECTOR
                                        951$:
                                                INC     R2                ;NEXT SECTOR
                                                SUB     #40.,R2           ;DONE WITH TRACK?
                                                CMP     R2,#10.           ;
                                                BEQ     952$              ;YES
                                                JMP     3$                ;NO
                                        952$:

                                                TST     TEMP1             ;WHICH SURFACE?
                                                BEQ     5$                ;TOP (0), BRANCH

                                                CLR     TEMP1             ;BOTTOM, SWITCH TO TOP WITH
                                                ADD     #200,R1
                                                MOV     #205,BDA(R4)      ;SEEK, GO IN ALSO
                                        4$:     MOV     #SEEK,FUNC(R4)    ;GO SEEK
                                                JSR     R5,LDFUNC
```

```
    164 023462  004537  022322                  JSR     R5,WTRDY
    165
    166 023466  000137  022652                  JMP     CONWR
    167
    168 023472  012737  000100  002226  5$:     MOV     #HEAD,TEMP1     ;WAS TOP, MAKE BOTTOM.
    169 023500  012764  000021  000040          MOV     #21,BDA(R4)
    170 023506  000760                          BR      4$
    171
    172
    173 023510  010146                  INBAD:  MOV     R1,-(SP)        ;SAVE R1
    174 023512  016403  000104                  MOV     DCS(R4),R3      ;GET THE CSR ADDRESS
    175 023516  016337  000000  002276          MOV     CS(R3),E.CS     ;GET THE ERROR INFO FROM CSR
    176 023524  016337  000002  002300          MOV     BA(R3),E.BA
    177 023532  016337  000004  002302          MOV     DA(R3),E.DA
    178 023540  004537  023630                  JSR     R5,GETSYS       ;UPDATE THE RUN TIME
    179 023544  004537  022372                  JSR     R5,GETDST       ;GET THE CURRENT DRIVE STATUS
    180 023550  010137  002304                  MOV     R1,E.MP         ;SAVE IT AS "(RLMP)" DATA
    181 023554                                  ERRHRD  199.,NWRTS,ERR12
        023554  104463                          TRAP    T$ERCODE
        023556  000307                          .WORD   199
        023560  002573                          .WORD   NWRTS
        023562  005304                          .WORD   ERR12
    182 023564  005264  000012                  INC     ERRCNT(R4)
    183 023570  005737  010310                  TST     T.DRP           ;ARE WE COUNTING ERRORS
    184 023574  001413                          BEQ     2$              ;NO
    185 023576  026437  000012  010250          CMP     ERRCNT(R4),ERLMT;PAST IT
    186 023604  103407                          BLO     2$              ;NO
    187 023606  012737  003157  002132          MOV     #ERLMTM,WHY
    188 023614  004537  021502                  JSR     R5,DRDRV
    189 023620  012705  022714                  MOV     #ENDWR,R5
    190
    191 023624  012601                  2$:     MOV     (SP)+,R1        ;RESET R1
    192 023626  000205                          RTS     R5
```

```
                                        .SBTTL  ROUTINE FOR SYSTEM CLOCK
                                        ;ROUTINE TO READ SYSTEM CLOCK
                                        ;USES 'REGTIM' FROM DIAGNOSTIC SUPERVISOR
023630  005737  002314          GETSYS: TST     SYSCLK          ;DO WE HAVE A CLOCK
023634  001002                          BNE     4$              ;YES, GO SERVICE IT
023636                                  BREAK                   ;NO, CALL SUPER FOR ^C
023636  104022                          EMT     C$BRK
023640  000205                          RTS     R5              ;EXIT
023642                          4$:     REQTIM  R0              ;GET PRESENT TIME
023642  104045                          EMT     C$REQTIM
023644  020037  002266          1$:     CMP     R0,LSTTIM       ;HAS IT MOVED
023650  001437                          BEQ     3$              ;NO MOVEMEMT SINCE LAST CALL
023652  013701  002266                  MOV     LSTTIM,R1       ;CALCULATE DIFFERENCE
023656  010037  002266                  MOV     R0,LSTTIM       ;AND FIX ACCORDINGLY
023662  160100                          SUB     R1,R0
023664  060037  002270          2$:     ADD     R0,SECOND       ;BUMP SECONDS
023670  022737  000074  002270        CMP     #60.,SECOND     ;SECONDS OVERFLOW
023676  003024                          BGT     3$
023700  162737  000074  002270  7$:     SUB     #60.,SECOND
023706  005237  002264                  INC     INTERVAL        ;TIME BETWEEN REPORTS
023712  005237  002272                  INC     MINUTE          ;BUMP MINUTES
023716  022737  000074  002270        CMP     #60.,SECOND
023724  003765                          BLE     7$
023726  022737  000074  002272        CMP     #60.,MINUTE
023734  003005                          BGT     3$
023736  005237  002274                  INC     HOUR
023742  162737  000074  002272        SUB     #60.,MINUTE
023750  000205                  3$:     RTS     R5
```

```
                                .SBTTL  HEADS HOME ROUTINE
023752                          STARS
                                ;;*****************************************************************
                                ;HDHOME -- ROUTINE TO BRING HEADS OVER TRACK 0
023752                          STARS
                                ;;*****************************************************************

023752  010046                  HDHOME: MOV     R0,-(SP)        ;SAVE R0
023754  012764  000010  000044          MOV     #RDHDR,FUNC(R4) ;READ HEADER
023762  004537  015226                  JSR     R5,LDFUNC       ;GO DO IT.
023766  004537  022322                  JSR     R5,WTRDY

023772  016300  000006                  MOV     MP(R3),R0       ;GET HEADER
023776  042700  000177                  BIC     #177,R0         ;ONLY CYLINDER
024002  010064  000050                  MOV     R0,LSTHDR(R4)   ;SAVE THIS CYL # AS THE LAST POSITION
024006  010064  000040                  MOV     R0,BDA(R4)      ;MOVE IT TO BUFFERED DA
024012  052764  000001  000040          BIS     #MK,BDA(R4)     ;SET MARKER FOR SEEK TO 000
024020  012764  000006  000044          MOV     #SEEK,FUNC(R4)  ;LOAD SEEK
024026  004537  015226                  JSR     R5,LDFUNC       ;SEEK!
024032  004537  022322                  JSR     R5,WTRDY        ;WAIT.
024036  005064  000124                  CLR     PRPOS(R4)       ;SET BUFFER TO HOME CYLINDER (000)
024042  012600                          MOV     (SP)+,R0
024044  000205                          RTS     R5
```

```
                        .SBTTL  RANDOM WC AND DA ROUTINE
024046                  STARS
                        ;;*****************************************************************
                        ;GWCDA -- 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
024046                  STARS
                        ;;*****************************************************************

024046  023737  010302  010304  GWCDA:  CMP     T.MXS,T.MNS     ;MIN MAX SECTORS EQUAL
024054  001003                          BNE     99$             ;NO, CALCULATE ONE
024056  013702  010302                  MOV     T.MXS,R2        ;LOAD SECTOR
024062  000421                          BR      5$              ;GO GET WC
024064  004537  022464          99$:    JSR     R5,RAND         ;GET RANDOM # FOR SECTOR
024070  013702  002146                  MOV     LONUM,R2
024074  042702  177700          1$:     BIC     #177700,R2      ;0-77 ONLY
024100  023702  010302                  CMP     T.MXS,R2        ;R2 LOWER THAN MAX
024104  103003                          BHIS    3$              ;BRANCH IF YES
024106  006202                          ASR     R2              ;HALF IT
024110  005202                          INC     R2              ;INC SO NOT 0
024112  000770                          BR      1$
024114  020237  010304          3$:     CMP     R2,T.MNS        ;MIN OKAY
024120  103002                          BHIS    5$
024122  006102                          ROL     R2
024124  000763                          BR      1$


                        ;NOW GET WORD COUNT

024126  005737  010336          5$:     TST     T.STIP          ;RESTRICT THE XFER SIZE?
024132  001003                          BNE     95$             ;BR IF YES
024134  013737  002322  010270          MOV     MAXWC,T.MXB     ;NO - MAKE MAXWC = BIGEST XFER SIZE AVAIL.
024142  023737  002322  010270  95$:    CMP     MAXWC,T.MXB
024150  103021                          BHIS    97$

024152                                  PRINTF  #FMT13D,#OVER,T.MXB,MAXWC
024152  013746  002322                  MOV     MAXWC,-(SP)
024156  013746  010270                  MOV     T.MXB,-(SP)
024162  012746  003310                  MOV     #OVER,-(SP)
024166  012746  007105                  MOV     #FMT13D,-(SP)
024172  012746  000004                  MOV     #4,-(SP)
024176  010600                          MOV     SP,R0
024200  104017                          EMT     C$PNTF
024202  062706  000012                  ADD     #12,SP
024206  013737  002322  010270          MOV     MAXWC,T.MXB

024214  023737  010270  010312  97$:    CMP     T.MXB,T.MNB     ;MIN MAX EQUAL
024222  003006                          BGT     6$
024224  013737  010270  010312          MOV     T.MXB,T.MNB

024232  013703  010270                  MOV     T.MXB,R3        ;YES SET WC
024236  000421                          BR      9$
024240  004537  022464          6$:     JSR     R5,RAND         ;GET RANDOM WORD COUNT
```

```
     024244  013703  002146              MOV     LONUM,R3
     024250  042703  160000      7$:     BIC     #160000,R3      ;MAX!!!!!
     024254  023703  010270              CMP     T.MXB,R3
     024260  103003                      BHIS    8$
     024262  006203                      ASR     R3
     024264  005203                      INC     R3
     024266  000770                      BR      7$
     024270  020337  010312      8$:     CMP     R3,T.MNB
     024274  103002                      BHIS    9$
     024276  006103                      ROL     R3
     024300  000763                      BR      7$

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

     024302  012701  000050      9$:     MOV     #40.,R1         ;SETUP FOR FOURTY SECTORS
     024306  005403                      NEG     R3              ;MAKE WORD COUNT NEGATIVE
     024310  010364  000042              MOV     R3,BMP(R4)      ;LOAD WORD COUNT
     024314  005301              11$:    DEC     R1              ;DOWN COUNT MINIMUM START SECT NEEDED
     024316  062703  000200              ADD     #128.,R3        ;ONE SECTOR'S WORTH
     024322  100774                      BMI     11$             ;STILL NEED ANOTHER SECTOR
     024324  020201                      CMP     R2,R1           ;DID RANDOM SECTOR SUFFICE
     024326  101401                      BLOS    12$             ;BRANCH IF SUFFICED
     024330  010102                      MOV     R1,R2           ;NO, THEN MAKE IT FIT
     024332  016464  000124  000040 12$:  MOV     PRPOS(R4),BDA(R4)
     024340  042764  000077  000040      BIC     #77,BDA(R4)
     024346  050264  000040              BIS     R2,BDA(R4)
     024352  000205                      RTS     R5
```

```
                                .SBTTL  ROUTINE TO DUMP BUFFER ON DCK
024354                          STARS
                                ;;*****************************************************************
                                ;DMPBUF -- 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.
024354                          STARS
                                ;;*****************************************************************

024354  004737  005716          DMPBUF: JSR     PC,LINE3

                                ;CALCULATE THE STARTING BUS ADDRESS FOR THE COMPARE

024360  012737  000200  002344          MOV     #128.,DWCNT1
024366  016400  000040                  MOV     BDA(R4),R0              ;GET STARTING BUS ADDRESS

024372  013701  002302                  MOV     E.DA,R1                 ;GET PRESENT DISK ADDRESS
024376  042700  177700                  BIC     #177700,R0              ;SAVE SECTOR BITS
024402  042701  177700                  BIC     #177700,R1              ;
024406  010002                          MOV     R0,R2                   ;SAVE A COPY
024410  010103                          MOV     R1,R3                   ;SAVE ANOTHER
024412  160203                          SUB     R2,R3                   ;GET DIFF OF SECTORS
024414  005002                          CLR     R2                      ;CALCULATE WORD COUNT
024416  062702  000200          93$:    ADD     #128.,R2                ;ONE SECTORS WORTH
024422  005303                          DEC     R3                      ;DONE
024424  001374                          BNE     93$                     ;NO
024426  016403  000042                  MOV     BMP(R4),R3              ;GET WORD COUNT
024432  005403                          NEG     R3                      ;MAKE IT POSITIVE
024434  020203                          CMP     R2,R3                   ;WORKING WITH FULL SECTOR
024436  003005                          BGT     94$                     ;NO, GO CALC PARTIAL SECTOR
024440  013702  002300                  MOV     E.BA,R2                 ;PRESENT BUS ADDRESS
024444  162702  000400                  SUB     #400,R2                 ;START OF COMPARE
024450  000412                          BR      96$                     ;GO COMPARE BUFFER
024452  160302                  94$:    SUB     R3,R2                   ;GET SECTOR DIFF
024454  012700  000200                  MOV     #128.,R0
024460  160200                          SUB     R2,R0
024462  010037  002344                  MOV     R0,DWCNT1
024466  006300                          ASL     R0
024470  013702  002300                  MOV     E.BA,R2
024474  160002                          SUB     R0,R2
024476                          96$:    PRINTB  #FMT13,#BUSAD,R2,#CRLDA,CHKSEC
024476  013746  002220                  MOV     CHKSEC,-(SP)
024502  012746  002441                  MOV     #CRLDA,-(SP)
024506  010246                          MOV     R2,-(SP)
024510  012746  003770                  MOV     #BUSAD,-(SP)
024514  012746  007070                  MOV     #FMT13,-(SP)
024520  012746  000005                  MOV     #5,-(SP)
024524  010600                          MOV     SP,R0
024526  104014                          EMT     C$PNTB
024530  062706  000014                  ADD     #14,SP
024534  012700  026066                  MOV     #PATLST,R0              ;CHECK PATTERN LIST
024540  012701  000010                  MOV     #8.,R1
024544  022062  000002          1$:     CMP     (R0)+,2(R2)
024550  001415                          BEQ     2$
024552  005301                          DEC     R1
024554  001373                          BNE     1$

```

```
 47 024556                              3$:     PRINTB  #FMT14,#NOREV
    024556  012746  003466                      MOV     #NOREV,-(SP)
    024562  012746  007132                      MOV     #FMT14,-(SP)
    024566  012746  000002                      MOV     #2,-(SP)
    024572  010600                              MOV     SP,R0
    024574  104014                              EMT     C$PNTB
    024576  062706  000006                      ADD     #6,SP
 48 024602  000532                              BR      STDMP
 49
 50 024604  021227  000200              2$:     CMP     (R2),#128.
 51 024610  101362                              BHI     3$
 52 024612  005037  002222                      CLR     DECNT
 53 024616  013701  010332                      MOV     T.CLT,R1
 54
 55 024622  012237  002224                      MOV     (R2)+,TEMP0             ;NONZERO WORD COUNT
 56 024626  013737  002224  002342              MOV     TEMP0,DWCNT
 57 024634  005437  002342                      NEG     DWCNT
 58 024640  012237  002226                      MOV     (R2)+,TEMP1
 59 024644  162737  000002  002224              SUB     #2,TEMP0
 60 024652  012737  000002  002230              MOV     #2,TEMP2                ;WORD
 61 024660  013703  002226                      MOV     TEMP1,R3                ;PATTERN ADDRESS
 62 024664  012737  000020  002236              MOV     #16.,TEMP5              ;16 ENTRIES
 63 024672  005737  002224              4$:     TST     TEMP0                   ;ZERO OR PATTERN
 64 024676  001417                              BEQ     6$                      ;ZERO BRANCH
 65 024700  005337  002224                      DEC     TEMP0
 66 024704  005737  002236                      TST     TEMP5                   ;WITHIN LIST
 67 024710  001005                              BNE     5$
 68 024712  012737  000020  002236              MOV     #16.,TEMP5
 69 024720  013703  002226                      MOV     TEMP1,R3
 70 024724  012337  002260              5$:     MOV     (R3)+,GDDAT
 71 024730  005337  002236                      DEC     TEMP5
 72 024734  000402                              BR      7$
 73 024736  005037  002260              6$:     CLR     GDDAT
 74 024742  005237  002342              7$:     INC     DWCNT
 75 024746  021237  002260                      CMP     (R2),GDDAT
 76 024752  001422                              BEQ     8$
 77
 78 024754  005237  002222                      INC     DECNT
 79 024760  005701                              TST     R1
 80 024762  001416                              BEQ     8$
 81 024764  005301                              DEC     R1
 82 024766                                      PRINTB  #FMT14B,TEMP2,GDDAT,(R2)
    024766  011246                              MOV     (R2),-(SP)
    024770  013746  002260                      MOV     GDDAT,-(SP)
    024774  013746  002230                      MOV     TEMP2,-(SP)
    025000  012746  007153                      MOV     #FMT14B,-(SP)
    025004  012746  000004                      MOV     #4,-(SP)
    025010  010600                              MOV     SP,R0
    025012  104014                              EMT     C$PNTB
    025014  062706  000012                      ADD     #12,SP
 83
 84 025020  005237  002230              8$:     INC     TEMP2
 85 025024  005722                              TST     (R2)+
 86 025026  023737  002230  002344              CMP     TEMP2,DWCNT1
 87 025034  003716                              BLE     4$
 88 025036                                      PRINTB  #FMT9A,DECNT,TEMP2
    025036  013746  002230                      MOV     TEMP2,-(SP)
```

```
        025042  013746  002222                  MOV     DECNT,-(SP)
        025046  012746  006651                  MOV     #FMT9A,-(SP)
        025052  012746  000003                  MOV     #3,-(SP)
        025056  010600                          MOV     SP,R0
        025060  104014                          EMT     C$PNTB
        025062  062706  000010                  ADD     #10,SP
    89
    90  025066  000205                          RTS     R5
    91
    92                                  ;ROUTINE TO DUMP THE CONTENTS OF THE READ BUFFER ON ERROR DETEDTED
    93                                  ;WILL ALSO TELL HOW MANY WORDS WERE IN THE XFER
    94
    95  025070  016437  000042  002224  STDMP:  MOV     BMP(R4),TEMP0           ;GET NEGATIVE WORD COUNT
    96  025076  005437  002224          NEG     TEMP0                   ;MAKE THE # POSITIVE
    97  025102  012737  000200  002344          MOV     #128.,DWCNT1            ;SET THE SIZE OF SECTOR
    98  025110                          PRINTB  #FMTXS,TEMP0            ;TELL TRANSFER SIZE
        025110  013746  002224          MOV     TEMP0,-(SP)
        025114  012746  007342          MOV     #FMTXS,-(SP)
        025120  012746  000002          MOV     #2,-(SP)
        025124  010600                  MOV     SP,R0
        025126  104014                  EMT     C$PNTB
        025130  062706  000006          ADD     #6,SP
    99  025134  013701  010332          MOV     T.CLT,R1                ;GET THE PRINT LIMIT
   100  025140  012703  000012          MOV     #10.,R3                 ;SETUP LINE LIMIT
   101  025144                  1$:     PRINTB  #FMT14A,(R2)            ;PRINT A DATA WORD
        025144  011246                  MOV     (R2),-(SP)
        025146  012746  007141          MOV     #FMT14A,-(SP)
        025152  012746  000002          MOV     #2,-(SP)
        025156  010600                  MOV     SP,R0
        025160  104014                  EMT     C$PNTB
        025162  062706  000006          ADD     #6,SP
   102  025166  005722                  TST     (R2)+                   ;POINT TO THE NEXT DATA WORD
   103  025170  005303                  DEC     R3                      ;DONE WITH THE LINE?
   104  025172  001012                  BNE     2$                      ;BR IF NO
   105  025174                          PRINTB  #FMT14C                 ;YES - PRINT <CR>
        025174  012746  007150          MOV     #FMT14C,-(SP)
        025200  012746  000001          MOV     #1,-(SP)
        025204  010600                  MOV     SP,R0
        025206  104014                  EMT     C$PNTB
        025210  062706  000004          ADD     #4,SP
   106  025214  012703  000012          MOV     #10.,R3                 ;RESET THE LINE LIMIT
   107  025220  005337  002344  2$:     DEC     DWCNT1                  ;END OF SECTOR?
   108  025224  001001                  BNE     3$                      ;BR IF NO
   109  025226  000402                  BR      4$                      ;YES - EXIT
   110  025230  005301          3$:     DEC     R1                      ;AT PRINT LIMIT?
   111  025232  001344                  BNE     1$                      ;BR IF NO
   112  025234                  4$:     PRINTB  #FMT14C                 ;PRINT <CR>
        025234  012746  007150          MOV     #FMT14C,-(SP)
        025240  012746  000001          MOV     #1,-(SP)
        025244  010600                  MOV     SP,R0
        025246  104014                  EMT     C$PNTB
        025250  062706  000004          ADD     #4,SP
   113  025254  000205                  RTS     R5                      ;EXIT
   114
   115
   116
   117                          ;
```

```
                                ;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
                                ;
                                CLEAR:  MOV     R4,-(SP)            ;SAVE R4
                                        MOV     #DRBUF,R4           ;GET BUFFER STARTS
                                2$:     CLR     (R4)+               ;CLEAR
                                        CMP     R4,#ENDBUF          ;AT END OF BUFFERS
                                        BNE     2$                  ;NO, GO TO 2$
                                4$:     MOV     (SP)+,R4            ;RESTORE CURRENT BUFFER POINTER
                                        RTS     R5                  ;EXIT
```

```
                                .SBTTL  ROUTINE TO CHECK FOR BAD SECTOR
025300                          STARS
                                ;;*****************************************************************
                                ;CKBDSC -- 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.
025300                          STARS
                                ;;*****************************************************************

025300  005037  002216  CKBDSC: CLR     HDRFND          ;CLEAR FLAG
025304  010046                  MOV     R0,-(SP)        ;SAVE R0
025306  010246                  MOV     R2,-(SP)        ;SAVE   R2
025310  012700  000021          MOV     #17.,R0         ;16 ENTRIES + BSF POINTER
025314  016402  000112  1$:     MOV     BSECPT(R4),R2   ;GET WHERE WE'RE LOOKING
025320  022712  177777  2$:     CMP     #-1,(R2)        ;END OF ENTRY LIST?
025324  001411                  BEQ     4$              ;BRANCH IF END
025326  023712  002220          CMP     CHKSEC,(R2)     ;HAVE WE GOT A MATCH
025332  001404                  BEQ     3$              ;THEN GO SET INDICATOR, ELSE
025334  005722                  TST     (R2)+
025336  005300                  DEC     R0
025340  001367                  BNE     2$
025342  000402                  BR      4$

025344  005237  002216  3$:     INC     HDRFND          ;SET FLAG FOUND

025350  012602  ..      4$:     MOV     (SP)+,R2
025352  012600                  MOV     (SP)+,R0
025354  000205                  RTS     R5

025356                          STARS
                                ;;*****************************************************************
                                ;CKBDTK -- HERE TO CHECK IF CYLINDER & HEAD SELECTED IS IN THE BAD SECTOR FILE
025356                          STARS
                                ;;*****************************************************************

025356  005037  002216  CKBDTK: CLR     HDRFND          ;CLEAR FLAG
025362  010046                  MOV     R0,-(SP)        ;SAVE R0
025364  010146                  MOV     R1,-(SP)        ;SAVE R1
025366  010246                  MOV     R2,-(SP)        ;SAVE R2
025370  012700  000021          MOV     #17.,R0         ;16 ENTRIES + BSF POINTER
025374  016402  000112  1$:     MOV     BSECPT(R4),R2   ;GET WHERE WE'RE LOOKING
025400  022712  177777  2$:     CMP     #-1,(R2)        ;END OF LIST?
025404  001414                  BEQ     4$              ;BRANCH IF END
025406  011201                  MOV     (R2),R1         ;GET THE ENTRY FROM BAD SECT FILE
025410  043701  002156          BIC     SMSK,R1         ;LEAVE ONLY CYL # & HEAD
025414  023701  002220          CMP     CHKSEC,R1       ;HAVE WE GOT A MATCH
025420  001404                  BEQ     3$              ;THEN GO SET INDICATOR, ELSE
025422  005722                  TST     (R2)+
025424  005300                  DEC     R0
025426  001364                  BNE     2$
025430  000402                  BR      4$

025432  005237  C02216  3$:     INC     HDRFND          ;SET FLAG FOUND

025436  012602          4$:     MOV     (SP)+,R2
025440  012601                  MOV     (SP)+,R1
025442  012600                  MOV     (SP)+,R0
025444  000205                  RTS     R5
```

```
                                        STARS
                                        ;;*****************************************************************
                                        STARS
                                        ;;*****************************************************************
                                        ;BUFFER TO STORE BAD SECTOR LISTS

025446                                  BSEC0:  .BLKW   17.
025510                                  BSEC1:  .BLKW   17.
025552                                  BSEC2:  .BLKW   17.
025614                                  BSEC3:  .BLKW   17.
025656                                  BSEC4:  .BLKW   17.
025720                                  BSEC5:  .BLKW   17.
025762                                  BSEC6:  .BLKW   17.
026024                                  BSEC7:  .BLKW   17.
026066                                  STARS
                                        ;;*****************************************************************
026066                                  STARS
                                        ;;*****************************************************************

                                        ;LIST OF PATTERNS USED IN WRITING

026066  026106                          PATLST: PAT0    ;ALL 0'S
026070  026146                                  PAT1    ;-1'S TO ALT BITS
026072  026206                                  PAT2    ;0'S TO ALT BITS
026074  026246                                  PAT3    ;SHIFTING ALT BITS
026076  026306                                  PAT4    ;WORST CASE DATA
026100  026346                                  PAT5    ;STRANGE DATA
026102  026406                                  PAT6    ;ALL 1'S
026104  026446                                  PAT7    ;STRANGE DATA

026106  000000                          PAT0:   .WORD   0
026110  000000                                  .WORD   0
026112  000000                                  .WORD   0
026114  000000                                  .WORD   0
026116  000000                                  .WORD   0
026120  000000                                  .WORD   0
026122  000000                                  .WORD   0
026124  000000                                  .WORD   0
026126  000000                                  .WORD   0
026130  000000                                  .WORD   0
026132  000000                                  .WORD   0
026134  000000                                  .WORD   0
026136  000000                                  .WORD   0
026140  000000                                  .WORD   0
026142  000000                                  .WORD   0
026144  000000                                  .WORD   0

026146  177777                          PAT1:   .WORD   177777
026150  177777                                  .WORD   177777
026152  177777                                  .WORD   177777
026154  052525                                  .WORD   052525
026156  052525                                  .WORD   052525
026160  052525                                  .WORD   052525
026162  177777                                  .WORD   177777
026164  177777                                  .WORD   177777
026166  052525                                  .WORD   052525
026170  052525                                  .WORD   052525
```

```
     54 026172  177777                        .WORD   177777
     55 026174  052525                        .WORD   052525
     56 026176  177252                        .WORD   177252
     57 026200  177252                        .WORD   177252
     58 026202  172765                        .WORD   172765
     59 026204  172765                        .WORD   172765
     60
     61 026206  000000              PAT2:     .WORD   0
     62 026210  000000                        .WORD   0
     63 026212  000000                        .WORD   0
     64 026214  177777                        .WORD   177777
     65 026216  177777                        .WORD   177777
     66 026220  177777                        .WORD   177777
     67 026222  000000                        .WORD   0
     68 026224  000000                        .WORD   0
     69 026226  177777                        .WORD   177777
     70 026230  177777                        .WORD   177777
     71 026232  000000                        .WORD   0
     72 026234  177777                        .WORD   177777
     73 026236  000000                        .WORD   0
     74 026240  177777                        .WORD   177777
     75 026242  000000                        .WORD   0
     76 026244  177777                        .WORD   177777
     77
     78 026246  025252              PAT3:     .WORD   25252
     79 026250  052525                        .WORD   52525
     80 026252  052525                        .WORD   52525
     81 026254  125252                        .WORD   125252
     82 026256  125252                        .WORD   125252
     83 026260  125252                        .WORD   125252
     84 026262  052525                        .WORD   52525
     85 026264  052525                        .WORD   52525
     86 026266  125252                        .WORD   125252
     87 026270  125252                        .WORD   125252
     88 026272  052525                        .WORD   52525
     89 026274  125252                        .WORD   125252
     90 026276  052525                        .WORD   52525
     91 026300  125252                        .WORD   125252
     92 026302  052525                        .WORD   52525
     93 026304  125252                        .WORD   125252
     94
     95 026306  155555              PAT4:     .WORD   155555
     96 026310  066666                        .WORD   066666
     97 026312  133333                        .WORD   133333
     98 026314  155555                        .WORD   155555
     99 026316  066666                        .WORD   066666
    100 026320  133333                        .WORD   133333
    101 026322  155555                        .WORD   155555
    102 026324  066666                        .WORD   066666
    103 026326  133333                        .WORD   133333
    104 026330  155555                        .WORD   155555
    105 026332  066666                        .WORD   066666
    106 026334  133333                        .WORD   133333
    107 026336  155555                        .WORD   155555
    108 026340  066666                        .WORD   066666
    109 026342  133333                        .WORD   133333
    110 026344  155555                        .WORD   155555
```

```
                111
026346  121105          PAT5:   .WORD   121105
026350  150442                  .WORD   150442
026352  064221                  .WORD   64221
026354  132110                  .WORD   132110
026356  055044                  .WORD   55044
026360  026422                  .WORD   26422
026362  013211                  .WORD   13211
026364  105504                  .WORD   105504
026366  042642                  .WORD   42642
026370  021321                  .WORD   21321
026372  110550                  .WORD   110550
026374  044264                  .WORD   44264
026376  022132                  .WORD   22132
026400  011055                  .WORD   11055
026402  104426                  .WORD   104426
026404  042213                  .WORD   42213
                128
026406  177777          PAT6:   .WORD   177777
026410  177777                  .WORD   177777
026412  177777                  .WORD   177777
026414  177777                  .WORD   177777
026416  177777                  .WORD   177777
026420  177777                  .WORD   177777
026422  177777                  .WORD   177777
026424  177777                  .WORD   177777
026426  177777                  .WORD   177777
026430  177777                  .WORD   177777
026432  177777                  .WORD   177777
026434  177777                  .WORD   177777
026436  177777                  .WORD   177777
026440  177777                  .WORD   177777
026442  177777                  .WORD   177777
026444  177777                  .WORD   177777
                145
026446  045513          PAT7:   .WORD   45513
026450  122645                  .WORD   122645
026452  151322                  .WORD   151322
026454  064551                  .WORD   64551
026456  132264                  .WORD   132264
026460  055132                  .WORD   55132
026462  026455                  .WORD   26455
026464  113226                  .WORD   113226
026466  045513                  .WORD   45513
026470  122645                  .WORD   122645
026472  151322                  .WORD   151322
026474  064551                  .WORD   64551
026476  132264                  .WORD   132264
026500  055132                  .WORD   55132
026502  026455                  .WORD   26455
026504  113226                  .WORD   113226
                162
                163
                164
026506  000240          ENDOFPROGRAM:   NOP
026510                  ENDTST
026510                  L10022:
```

```
026510  104001                      EMT     C$ETST
026512  000000                      HALT

                        .SBTTL  DRIVE INFORMATION BUFFERS

                        ;DRIVE INFORMATION BUFFER


                        .LIST   ME

026514                  DRBUF:
        000010          .REPT   8.

026514  000000                      SKCNT                   ;SEEK OPERATION COUNT
026516  000002                      RXFR1                   ;READ OPERATION COUNT (BITS) LOW ORDER
026520  000004                      RXFR2                   ; "       "        "     "   HIGH ORDER
026522  000006                      WXFR1                   ;WRITE OPERATION COUNT (BITS) LOW ORDER
026524  000010                      WXFR2                   ; "       "        "     "   HIGH ORDER
026526  000012                      ERRCNT                  ;ERROR COUNT - HARD
026530  000014                      SFTCNT                  ;ERROR COUNT - SOFT
026532  000016                      SKECNT                  ;SEEK ERROR COUNT
026534  000020                      DERCNT                  ;DRIVE ERROR COUNT
026536  000022                      DCRCER                  ;DATA CRC ERROR COUNT
026540  000024                      HCRCER                  ;HEADER CRC ERROR COUNT
026542  000026                      DLTCNT                  ;DATA LATE ERROR COUNT
026544  000030                      OPICNT                  ;OPERATION INCOMPLETE ERROR COUNT
026546  000032                      HNFERR                  ;HEADER NOT FOUND ERROR COUNT
026550  000034                      NXMCNT                  ;NON EXISTANT MEMORY ERROR COUNT
026552  000036                      RETRY                   ;PRESENT RETRY NUMBER
026554  000040                      BDA                     ;    "      DISK ADDRESS CONTENTS
026556  000042                      BMP                     ;PRESENT MULTIPURPOSE CONTENTS
026560  000044                      FUNC                    ;LAST FUNCTION LOADED
026562  000046                      BCSADR                  ;CSR IMAGE OF LAST COMMAND
026564  000050                      LSTHDR                  ;LAST POSITION ON DISK
026566  000052                      RTYPE                   ;ERROR ON WHICH RECOVERY IS IN PROGRESS
026570  000054                      SKCNT1                  ;SEEK COUNT LOW ORDER
026572  000056                      PRFLGS                  ;PROGRAM INTERNAL FLAGS
026574  000060                      RXFR3                   ;READ COUNT THIRD
026576  000062                      WXFR3                   ;WRITE COUNT THIRD
026600  000064                      LSTDA                   ;DISK ADDRESS OF SOFT ERROR
026602  000066                      DIFWD                   ;LAST DIFFERENCE WORD OF SEEK
026604  000070                      DPHOUR                  ;TIME DRIVE WAS DROPPED
026606  000072                      TRERR                   ;TRACKING ERROR COUNT
026610  000074                      DATCER
026612  000076                      DOWCK                   ;WRITE CHECK NECESARY
026614  000100                      SERNM1                  ;SERIAL NUMBER OF CARTRIDGE
026616  000102                      SERNM2                  ;SERIAL NUMBER OF CARTRIDGE
026620  000104                      DCS                     ;CSR ADDRESS
026622  000106                      DRSEL                   ;DRIVE SELECT BITS(8,9,10)
026624  000110                      BBA                     ;PRESENT BUS ADDRESS CONTENTS
026626  000112                      BSECPT                  ;POINTER TO BAD SECTOR FILE
026630  000114                      RSEEK
026632  000116                      SOFTCS                  ;CSR AT TIME OF SOFT ERROR
026634  000120                      TDR                     ;DRIVE TYPE FLAG (RL02 =1)
026636  000122                      WRIPG                   ;WRITE IN PROGRESS FLAG
026640  000124                      PRPOS                   ;PRESENT POSITION ON DISK
```

```
026642  000000          SKCNT           ;SEEK OPERATION COUNT
026644  000002          RXFR1           ;READ OPERATION COUNT (BITS) LOW ORDER
026646  000004          RXFR2           ;  "         "        "    "    HIGH ORDER
026650  000006          WXFR1           ;WRITE OPERATION COUNT (BITS) LOW ORDER
026652  000010          WXFR2           ;  "         "        "    "    HIGH ORDER
026654  000012          ERRCNT          ;ERROR COUNT - HARD
026656  000014          SFTCNT          ;ERROR COUNT - SOFT
026660  000016          SKECNT          ;SEEK ERROR COUNT
026662  000020          DERCNT          ;DRIVE ERROR COUNT
026664  000022          DCRCER          ;DATA CRC ERROR COUNT
026666  000024          HCRCER          ;HEADER CRC ERROR COUNT
026670  000026          DLTCNT          ;DATA LATE ERROR COUNT
026672  000030          OPICNT          ;OPERATION INCOMPLETE ERROR COUNT
026674  000032          HNFERR          ;HEADER NOT FOUND ERROR COUNT
026676  000034          NXMCNT          ;NON EXISTANT MEMORY ERROR COUNT
026700  000036          RETRY           ;PRESENT RETRY NUMBER
026702  000040          BDA             ;   "     DISK ADDRESS CONTENTS
026704  000042          BMP             ;PRESENT MULTIPURPOSE CONTENTS
026706  000044          FUNC            ;LAST FUNCTION LOADED
026710  000046          BCSADR          ;CSR IMAGE OF LAST COMMAND
026712  000050          LSTHDR          ;LAST POSITION ON DISK
026714  000052          RTYPE           ;ERROR ON WHICH RECOVERY IS IN PROGRESS
026716  000054          SKCNT1          ;SEEK COUNT LOW ORDER
026720  000056          PRFLGS          ;PROGRAM INTERNAL FLAGS
026722  000060          RXFR3           ;READ COUNT THIRD
026724  000062          WXFR3           ;WRITE COUNT THIRD
026726  000064          LSTDA           ;DISK ADDRESS OF SOFT ERROR
026730  000066          DIFWD           ;LAST DIFFERENCE WORD OF SEEK
026732  000070          DPHOUR          ;TIME DRIVE WAS DROPPED
026734  000072          TRERR           ;TRACKING ERROR COUNT
026736  000074          DATCER
026740  000076          DOWCK           ;WRITE CHECK NECESARY
026742  000100          SERNM1          ;SERIAL NUMBER OF CARTRIDGE
026744  000102          SERNM2          ;SERIAL NUMBER OF CARTRIDGE
026746  000104          DCS             ;CSR ADDRESS
026750  000106          DRSEL           ;DRIVE SELECT BITS(8,9,10)
026752  000110          BBA             ;PRESENT BUS ADDRESS CONTENTS
026754  000112          BSECPT          ;POINTER TO BAD SECTOR FILE
026756  000114          RSEEK
026760  000116          SOFTCS          ;CSR AT TIME OF SOFT ERROR
026762  000120          TDR             ;DRIVE TYPE FLAG (RL02 =1)
026764  000122          WRIPG           ;WRITE IN PROGRESS FLAG
026766  000124          PRPOS           ;PRESENT POSITION ON DISK

026770  000000          SKCNT           ;SEEK OPERATION COUNT
026772  000002          RXFR1           ;READ OPERATION COUNT (BITS) LOW ORDER
026774  000004          RXFR2           ;  "         "        "    "    HIGH ORDER
026776  000006          WXFR1           ;WRITE OPERATION COUNT (BITS) LOW ORDER
027000  000010          WXFR2           ;  "         "        "    "    HIGH ORDER
027002  000012          ERRCNT          ;ERROR COUNT - HARD
027004  000014          SFTCNT          ;ERROR COUNT - SOFT
027006  000016          SKECNT          ;SEEK ERROR COUNT
027010  000020          DERCNT          ;DRIVE ERROR COUNT
027012  000022          DCRCER          ;DATA CRC ERROR COUNT
027014  000024          HCRCER          ;HEADER CRC ERROR COUNT
027016  000026          DLTCNT          ;DATA LATE ERROR COUNT
027020  000030          OPICNT          ;OPERATION INCOMPLETE ERROR COUNT
```

```
027022  000032          HNFERR          ;HEADER NOT FOUND ERROR COUNT
027024  000034          NXMCNT          ;NON EXISTANT MEMORY ERROR COUNT
027026  000036          RETRY           ;PRESENT RETRY NUMBER
027030  000040          BDA             ;   ''      DISK ADDRESS CONTENTS
027032  000042          BMP             ;PRESENT MULTIPURPOSE CONTENTS
027034  000044          FUNC            ;LAST FUNCTION LOADED
027036  000046          BCSADR          ;CSR IMAGE OF LAST COMMAND
027040  000050          LSTHDR          ;LAST POSITION ON DISK
027042  000052          RTYPE           ;ERROR ON WHICH RECOVERY IS IN PROGRESS
027044  000054          SKCNT1          ;SEEK COUNT LOW ORDER
027046  000056          PRFLGS          ;PROGRAM INTERNAL FLAGS
027050  000060          RXFR3           ;READ COUNT THIRD
027052  000062          WXFR3           ;WRITE COUNT THIRD
027054  000064          LSTDA           ;DISK ADDRESS OF SOFT ERROR
027056  000066          DIFWD           ;LAST DIFFERENCE WORD OF SEEK
027060  000070          DPHOUR          ;TIME DRIVE WAS DROPPED
027062  000072          TRERR           ;TRACKING ERROR COUNT
027064  000074          DATCER
027066  000076          DOWCK           ;WRITE CHECK NECESARY
027070  000100          SERNM1          ;SERIAL NUMBER OF CARTRIDGE
027072  000102          SERNM2          ;SERIAL NUMBER OF CARTRIDGE
027074  000104          DCS             ;CSR ADDRESS
027076  000106          DRSEL           ;DRIVE SELECT BITS(8,9,10)
027100  000110          BBA             ;PRESENT BUS ADDRESS CONTENTS
027102  000112          BSECPT          ;POINTER TO BAD SECTOR FILE
027104  000114          RSEEK
027106  000116          SOFTCS          ;CSR AT TIME OF SOFT ERROR
027110  000120          TDR             ;DRIVE TYPE FLAG (RL02 =1)
027112  000122          WRIPG           ;WRITE IN PROGRESS FLAG
027114  000124          PRPOS           ;PRESENT POSITION ON DISK

027116  000000          SKCNT           ;SEEK OPERATION COUNT
027120  000002          RXFR1           ;READ OPERATION COUNT (BITS) LOW ORDER
027122  000004          RXFR2           ; ''      ''        ''    ''  HIGH ORDER
027124  000006          WXFR1           ;WRITE OPERATION COUNT (BITS) LOW ORDER
027126  000010          WXFR2           ; ''      ''        ''    ''  HIGH ORDER
027130  000012          ERRCNT          ;ERROR COUNT - HARD
027132  000014          SFTCNT          ;ERROR COUNT - SOFT
027134  000016          SKECNT          ;SEEK ERROR COUNT
027136  000020          DERCNT          ;DRIVE ERROR COUNT
027140  000022          DCRCER          ;DATA CRC ERROR COUNT
027142  000024          HCRCER          ;HEADER CRC ERROR COUNT
027144  000026          DLTCNT          ;DATA LATE ERROR COUNT
027146  000030          OPICNT          ;OPERATION INCOMPLETE ERROR COUNT
027150  000032          HNFERR          ;HEADER NOT FOUND ERROR COUNT
027152  000034          NXMCNT          ;NON EXISTANT MEMORY ERROR COUNT
027154  000036          RETRY           ;PRESENT RETRY NUMBER
027156  000040          BDA             ;   ''      DISK ADDRESS CONTENTS
027160  000042          BMP             ;PRESENT MULTIPURPOSE CONTENTS
027162  000044          FUNC            ;LAST FUNCTION LOADED
027164  000046          BCSADR          ;CSR IMAGE OF LAST COMMAND
027166  000050          LSTHDR          ;LAST POSITION ON DISK
027170  000052          RTYPE           ;ERROR ON WHICH RECOVERY IS IN PROGRESS
027172  000054          SKCNT1          ;SEEK COUNT LOW ORDER
027174  000056          PRFLGS          ;PROGRAM INTERNAL FLAGS
027176  000060          RXFR3           ;READ COUNT THIRD
027200  000062          WXFR3           ;WRITE COUNT THIRD
```

```
027202  000064          LSTDA           ;DISK ADDRESS OF SOFT ERROR
027204  000066          DIFWD           ;LAST DIFFERENCE WORD OF SEEK
027206  000070          DPHOUR          ;TIME DRIVE WAS DROPPED
027210  000072          TRERR           ;TRACKING ERROR COUNT
027212  000074          DATCER
027214  000076          DOWCK           ;WRITE CHECK NECESARY
027216  000100          SERNM1          ;SERIAL NUMBER OF CARTRIDGE
027220  000102          SERNM2          ;SERIAL NUMBER OF CARTRIDGE
027222  000104          DCS             ;CSR ADDRESS
027224  000106          DRSEL           ;DRIVE SELECT BITS(8,9,10)
027226  000110          BBA             ;PRESENT BUS ADDRESS CONTENTS
027230  000112          BSECPT          ;POINTER TO BAD SECTOR FILE
027232  000114          RSEEK
027234  000116          SOFTCS          ;CSR AT TIME OF SOFT ERROR
027236  000120          TDR             ;DRIVE TYPE FLAG (RL02 =1)
027240  000122          WRIPG           ;WRITE IN PROGRESS FLAG
027242  000124          PRPOS           ;PRESENT POSITION ON DISK

027244  000000          SKCNT           ;SEEK OPERATION COUNT
027246  000002          RXFR1           ;READ OPERATION COUNT (BITS) LOW ORDER
027250  000004          RXFR2           ;  "      "      "      "    HIGH ORDER
027252  000006          WXFR1           ;WRITE OPERATION COUNT (BITS) LOW ORDER
027254  000010          WXFR2           ;  "      "      "      "    HIGH ORDER
027256  000012          ERRCNT          ;ERROR COUNT - HARD
027260  000014          SFTCNT          ;ERROR COUNT - SOFT
027262  000016          SKECNT          ;SEEK ERROR COUNT
027264  000020          DERCNT          ;DRIVE ERROR COUNT
027266  000022          DCRCER          ;DATA CRC ERROR COUNT
027270  000024          HCRCER          ;HEADER CRC ERROR COUNT
027272  000026          DLTCNT          ;DATA LATE ERROR COUNT
027274  000030          OPICNT          ;OPERATION INCOMPLETE ERROR COUNT
027276  000032          HNFERR          ;HEADER NOT FOUND ERROR COUNT
027300  000034          NXMCNT          ;NON EXISTANT MEMORY ERROR COUNT
027302  000036          RETRY           ;PRESENT RETRY NUMBER
027304  000040          BDA             ;   "     DISK ADDRESS CONTENTS
027306  000042          BMP             ;PRESENT MULTIPURPOSE CONTENTS
027310  000044          FUNC            ;LAST FUNCTION LOADED
027312  000046          BCSADR          ;CSR IMAGE OF LAST COMMAND
027314  000050          LSTHDR          ;LAST POSITION ON DISK
027316  000052          RTYPE           ;ERROR ON WHICH RECOVERY IS IN PROGRESS
027320  000054          SKCNT1          ;SEEK COUNT LOW ORDER
027322  000056          PRFLGS          ;PROGRAM INTERNAL FLAGS
027324  000060          RXFR3           ;READ COUNT THIRD
027326  000062          WXFR3           ;WRITE COUNT THIRD
027330  000064          LSTDA           ;DISK ADDRESS OF SOFT ERROR
027332  000066          DIFWD           ;LAST DIFFERENCE WORD OF SEEK
027334  000070          DPHOUR          ;TIME DRIVE WAS DROPPED
027336  000072          TRERR           ;TRACKING ERROR COUNT
027340  000074          DATCER
027342  000076          DOWCK           ;WRITE CHECK NECESARY
027344  000100          SERNM1          ;SERIAL NUMBER OF CARTRIDGE
027346  000102          SERNM2          ;SERIAL NUMBER OF CARTRIDGE
027350  000104          DCS             ;CSR ADDRESS
027352  000106          DRSEL           ;DRIVE SELECT BITS(8,9,10)
027354  000110          BBA             ;PRESENT BUS ADDRESS CONTENTS
027356  000112          BSECPT          ;POINTER TO BAD SECTOR FILE
027360  000114          RSEEK
```

```
027362  000116                  SOFTCS          ;CSR AT TIME OF SOFT ERROR
027364  000120                  TDR             ;DRIVE TYPE FLAG (RL02 =1)
027366  000122                  WRIPG           ;WRITE IN PROGRESS FLAG
027370  000124                  PRPOS           ;PRESENT POSITION ON DISK

027372  000000                  SKCNT           ;SEEK OPERATION COUNT
027374  000002                  RXFR1           ;READ OPERATION COUNT (BITS) LOW ORDER
027376  000004                  RXFR2           ;  "         "         "    "   HIGH ORDER
027400  000006                  WXFR1           ;WRITE OPERATION COUNT (BITS) LOW ORDER
027402  000010                  WXFR2           ;  "          "         "    "   HIGH ORDER
027404  000012                  ERRCNT          ;ERROR COUNT - HARD
027406  000014                  SFTCNT          ;ERROR COUNT - SOFT
027410  000016                  SKECNT          ;SEEK ERROR COUNT
027412  000020                  DERCNT          ;DRIVE ERROR COUNT
027414  000022                  DCRCER          ;DATA CRC ERROR COUNT
027416  000024                  HCRCER          ;HEADER CRC ERROR COUNT
027420  000026                  DLTCNT          ;DATA LATE ERROR COUNT
027422  000030                  OPICNT          ;OPERATION INCOMPLETE ERROR COUNT
027424  000032                  HNFERR          ;HEADER NOT FOUND ERROR COUNT
027426  000034                  NXMCNT          ;NON EXISTANT MEMORY ERROR COUNT
027430  000036                  RETRY           ;PRESENT RETRY NUMBER
027432  000040                  BDA             ;  "     DISK ADDRESS CONTENTS
027434  000042                  BMP             ;PRESENT MULTIPURPOSE CONTENTS
027436  000044                  FUNC            ;LAST FUNCTION LOADED
027440  000046                  BCSADR          ;CSR IMAGE OF LAST COMMAND
027442  000050                  LSTHDR          ;LAST POSITION ON DISK
027444  000052                  RTYPE           ;ERROR ON WHICH RECOVERY IS IN PROGRESS
027446  000054                  SKCNT1          ;SEEK COUNT LOW ORDER
027450  000056                  PRFLGS          ;PROGRAM INTERNAL FLAGS
027452  000060                  RXFR3           ;READ COUNT THIRD
027454  000062                  WXFR3           ;WRITE COUNT THIRD
027456  000064                  LSTDA           ;DISK ADDRESS OF SOFT ERROR
027460  000066                  DIFWD           ;LAST DIFFERENCE WORD OF SEEK
027462  000070                  DPHOUR          ;TIME DRIVE WAS DROPPED
027464  000072                  TRERR           ;TRACKING ERROR COUNT
027466  000074                  DATCER
027470  000076                  DOWCK           ;WRITE CHECK NECESARY
027472  000100                  SERNM1          ;SERIAL NUMBER OF CARTRIDGE
027474  000102                  SERNM2          ;SERIAL NUMBER OF CARTRIDGE
027476  000104                  DCS             ;CSR ADDRESS
027500  000106                  DRSEL           ;DRIVE SELECT BITS(8,9,10)
027502  000110                  BBA             ;PRESENT BUS ADDRESS CONTENTS
027504  000112                  BSECPT          ;POINTER TO BAD SECTOR FILE
027506  000114                  RSEEK
027510  000116                  SOFTCS          ;CSR AT TIME OF SOFT ERROR
027512  000120                  TDR             ;DRIVE TYPE FLAG (RL02 =1)
027514  000122                  WRIPG           ;WRITE IN PROGRESS FLAG
027516  000124                  PRPOS           ;PRESENT POSITION ON DISK

027520  000000                  SKCNT           ;SEEK OPERATION COUNT
027522  000002                  RXFR1           ;READ OPERATION COUNT (BITS) LOW ORDER
027524  000004                  RXFR2           ;  "         "         "    "   HIGH ORDER
027526  000006                  WXFR1           ;WRITE OPERATION COUNT (BITS) LOW ORDER
027530  000010                  WXFR2           ;  "          "         "    "   HIGH ORDER
027532  000012                  ERRCNT          ;ERROR COUNT - HARD
027534  000014                  SFTCNT          ;ERROR COUNT - SOFT
027536  000016                  SKECNT          ;SEEK ERROR COUNT
```

```
027540  000020          DERCNT          ;DRIVE ERROR COUNT
027542  000022          DCRCER          ;DATA CRC ERROR COUNT
027544  000024          HCRCER          ;HEADER CRC ERROR COUNT
027546  000026          DLTCNT          ;DATA LATE ERROR COUNT
027550  000030          OPICNT          ;OPERATION INCOMPLETE ERROR COUNT
027552  000032          HNFERR          ;HEADER NOT FOUND ERROR COUNT
027554  000034          NXMCNT          ;NON EXISTANT MEMORY ERROR COUNT
027556  000036          RETRY           ;PRESENT RETRY NUMBER
027560  000040          BDA             ;   "     DISK ADDRESS CONTENTS
027562  000042          BMP             ;PRESENT MULTIPURPOSE CONTENTS
027564  000044          FUNC            ;LAST FUNCTION LOADED
027566  000046          BCSADR          ;CSR IMAGE OF LAST COMMAND
027570  000050          LSTHDR          ;LAST POSITION ON DISK
027572  000052          RTYPE           ;ERROR ON WHICH RECOVERY IS IN PROGRESS
027574  000054          SKCNT1          ;SEEK COUNT LOW ORDER
027576  000056          PRFLGS          ;PROGRAM INTERNAL FLAGS
027600  000060          RXFR3           ;READ COUNT THIRD
027602  000062          WXFR3           ;WRITE COUNT THIRD
027604  000064          LSTDA           ;DISK ADDRESS OF SOFT ERROR
027606  000066          DIFWD           ;LAST DIFFERENCE WORD OF SEEK
027610  000070          DPHOUR          ;TIME DRIVE WAS DROPPED
027612  000072          TRERR           ;TRACKING ERROR COUNT
027614  000074          DATCER
027616  000076          DOWCK           ;WRITE CHECK NECESARY
027620  000100          SERNM1          ;SERIAL NUMBER OF CARTRIDGE
027622  000102          SERNM2          ;SERIAL NUMBER OF CARTRIDGE
027624  000104          DCS             ;CSR ADDRESS
027626  000106          DRSEL           ;DRIVE SELECT BITS(8,9,10)
027630  000110          BBA             ;PRESENT BUS ADDRESS CONTENTS
027632  000112          BSECPT          ;POINTER TO BAD SECTOR FILE
027634  000114          RSEEK
027636  000116          SOFTCS          ;CSR AT TIME OF SOFT ERROR
027640  000120          TDR             ;DRIVE TYPE FLAG (RL02 =1)
027642  000122          WRIPG           ;WRITE IN PROGRESS FLAG
027644  000124          PRPOS           ;PRESENT POSITION ON DISK

027646  000000          SKCNT           ;SEEK OPERATION COUNT
027650  000002          RXFR1           ;READ OPERATION COUNT (BITS) LOW ORDER
027652  000004          RXFR2           ;   "       "       "    "   HIGH ORDER
027654  000006          WXFR1           ;WRITE OPERATION COUNT (BITS) LOW ORDER
027656  000010          WXFR2           ;   "       "       "    "   HIGH ORDER
027660  000012          ERRCNT          ;ERROR COUNT - HARD
027662  000014          SFTCNT          ;ERROR COUNT - SOFT
027664  000016          SKECNT          ;SEEK ERROR COUNT
027666  000020          DERCNT          ;DRIVE ERROR COUNT
027670  000022          DCRCER          ;DATA CRC ERROR COUNT
027672  000024          HCRCER          ;HEADER CRC ERROR COUNT
027674  000026          DLTCNT          ;DATA LATE ERROR COUNT
027676  000030          OPICNT          ;OPERATION INCOMPLETE ERROR COUNT
027700  000032          HNFERR          ;HEADER NOT FOUND ERROR COUNT
027702  000034          NXMCNT          ;NON EXISTANT MEMORY ERROR COUNT
027704  000036          RETRY           ;PRESENT RETRY NUMBER
027706  000040          BDA             ;   "     DISK ADDRESS CONTENTS
027710  000042          BMP             ;PRESENT MULTIPURPOSE CONTENTS
027712  000044          FUNC            ;LAST FUNCTION LOADED
027714  000046          BCSADR          ;CSR IMAGE OF LAST COMMAND
027716  000050          LSTHDR          ;LAST POSITION ON DISK
```

```
027720  000052                  RTYPE           ;ERROR ON WHICH RECOVERY IS IN PROGRESS
027722  000054                  SKCNT1          ;SEEK COUNT LOW ORDER
027724  000056                  PRFLGS          ;PROGRAM INTERNAL FLAGS
027726  000060                  RXFR3           ;READ COUNT THIRD
027730  000062                  WXFR3           ;WRITE COUNT THIRD
027732  000064                  LS1DA           ;DISK ADDRESS OF SOFT ERROR
027734  000066                  DIFWD           ;LAST DIFFERENCE WORD OF SEEK
027736  000070                  DPHOUR          ;TIME DRIVE WAS DROPPED
027740  000072                  TRERR           ;TRACKING ERROR COUNT
027742  000074                  DATCER
027744  000076                  DOWCK           ;WRITE CHECK NECESARY
027746  000100                  SERNM1          ;SERIAL NUMBER OF CARTRIDGE
027750  000102                  SERNM2          ;SERIAL NUMBER OF CARTRIDGE
027752  000104                  DCS             ;CSR ADDRESS
027754  000106                  DRSEL           ;DRIVE SELECT BITS(8,9,10)
027756  000110                  BBA             ;PRESENT BUS ADDRESS CONTENTS
027760  000112                  BSECPT          ;POINTER TO BAD SECTOR FILE
027762  000114                  RSEEK
027764  000116                  SOFTCS          ;CSR AT TIME OF SOFT ERROR
027766  000120                  TDR             ;DRIVE TYPE FLAG (RL02 =1)
027770  000122                  WRIPG           ;WRITE IN PROGRESS FLAG
027772  000124                  PRPOS           ;PRESENT POSITION ON DISK
  223                   .NLIST  ME
  224
  225 027774  000000    ENDBUF: .WORD   0
  226
  227 027776            BGNMOD  HRDPRM
  228 027776                    BGNHRD
      027776  000030            .WORD L10024-L$HARD/2
  229
  230 030000                    GPRML   CNTYPE,CNT,1,YES
      030000  005130            .WORD   T$CODE
      030002  030060            .WORD   CNTYPE
      030004  000001            .WORD   1
  231 030006                    GPRMA   CSRMSG,CSR,0,160000,177776,YES
      030006  000031            .WORD   T$CODE
      030010  030065            .WORD   CSRMSG
      030012  160000            .WORD   T$LOLIM
      030014  177776            .WORD   T$HILIM
  232 030016                    GPRMA   VECMSG,VECT,0,0,776,YES
      030016  001031            .WORD   T$CODE
      030020  030134            .WORD   VECMSG
      030022  000000            .WORD   T$LOLIM
      030024  000776            .WORD   T$HILIM
  233 030026                    GPRMD   DRMSG,DRBT,0,03400,0,7,YES
      030026  004032            .WORD   T$CODE
      030030  030143            .WORD   DRMSG
      030032  003400            .WORD   03400
      030034  000000            .WORD   T$LOLIM
      030036  000007            .WORD   T$HILIM
  234 030040                    GPRML   DRTYPE,TYPDR,1,YES
      030040  003130            .WORD   T$CODE
      030042  030112            .WORD   DRTYPE
      030044  000001            .WORD   1
  235 030046                    GPRMD   BRMSG,PRIOR,0,340,0,7,YES
      030046  002032            .WORD   T$CODE
      030050  030101            .WORD   BRMSG
```

```
        030052  000340                          .WORD   340
        030054  000000                          .WORD   T$LOLIM
        030056  000007                          .WORD   T$HILIM
    236
    237 030060                                  ENDHRD
                                                .EVEN
        030060                          L10024:
    238
    242
    243 030060      122     114     061 CNTYPE: .ASCIZ  /RL11/
    244 030065      102     125     123 CSRMSG: .ASCIZ  /BUS ADDRESS/
    245 030101      102     122     040 BRMSG:  .ASCIZ  /BR LEVEL/
    246 030112      104     122     111 DRTYPE: .ASCIZ  /DRIVE TYPE = RL01/
    247 030134      126     105     103 VECMSG: .ASCIZ  /VECTOR/
    248 030143      104     122     111 DRMSG:  .ASCIZ  /DRIVE/
    249
    253
    254                             .EVEN
    255
    256 030152                                  ENDMOD
    257
    258 030152                      BGNMOD      SFTPRM
    259
    260 030152                                  BGNSFT
        030152  000220                          .WORD L10025-L$SOFT/2
    261
    262 030154                                  GPRMD   RTMSG,RLT,D,177777,0,177777,YES
        030154  000052                          .WORD   T$CODE
        030156  031061                          .WORD   RTMSG
        030160  177777                          .WORD   177777
        030162  000000                          .WORD   T$LOLIM
        030164  177777                          .WORD   T$HILIM
    263 030166                                  GPRMD   SRTMSG,SRLT,D,177777,0,177777,YES
        030166  031052                          .WORD   T$CODE
        030170  030704                          .WORD   SRTMSG
        030172  177777                          .WORD   177777
        030174  000000                          .WORD   T$LOLIM
        030176  177777                          .WORD   T$HILIM
    264 030200                                  GPRML   FDCHK,DCKFG,1,YES
        030200  020130                          .WORD   T$CODE
        030202  031410                          .WORD   FDCHK
        030204  000001                          .WORD   1
    265 030206                                  XFERF   5$
        030206  006044                          .WORD   T$CODE
    266 030210                                  GPRMD   CHKLMT,CLMT,D,177777,0,128.,YES
        030210  032052                          .WORD   T$CODE
        030212  030723                          .WORD   CHKLMT
        030214  177777                          .WORD   177777
        030216  000000                          .WORD   T$LOLIM
        030220  000200                          .WORD   T$HILIM
    267 030222                          5$:     GPRMD   INMSG,TYT,D,177777,1,177777,YES
        030222  005052                          .WORD   T$CODE
        030224  031171                          .WORD   INMSG
        030226  177777                          .WORD   177777
        030230  000001                          .WORD   T$LOLIM
        030232  177777                          .WORD   T$HILIM
    268 030234                                  GPRML   DRPMS,DRFLG,1,YES
```

```
        030234  021130                  .WORD   T$CODE
        030236  031471                  .WORD   DRPMS
        030240  000001                  .WORD   1
    269 030242                          XFERF   3$
        030242  032044                  .WORD   T$CODE
    270 030244                          GPRMD   ERMSG,ELT,D,177777,0,177777,YES
        030244  001052                  .WORD   T$CODE
        030246  030775                  .WORD   ERMSG
        030250  177777                  .WORD   177777
        030252  000000                  .WORD   T$LOLIM
        030254  177777                  .WORD   T$HILIM
    271 030256                          GPRMD   SFTMSG,SEL,D,177777,0,177777,YES
        030256  023052                  .WORD   T$CODE
        030260  031011                  .WORD   SFTMSG
        030262  177777                  .WORD   177777
        030264  000000                  .WORD   T$LOLIM
        030266  177777                  .WORD   T$HILIM
    272 030270                          GPRMD   DERPMS,DCD,D,177777,0,177777,YES
        030270  036052                  .WORD   T$CODE
        030272  031525                  .WORD   DERPMS
        030274  177777                  .WORD   177777
        030276  000000                  .WORD   T$LOLIM
        030300  177777                  .WORD   T$HILIM
    273 030302                          GPRMD   SEMSG,SET,D,177777,0,177777,YES
        030302  002052                  .WORD   T$CODE
        030304  031073                  .WORD   SEMSG
        030306  177777                  .WORD   177777
        030310  000000                  .WORD   T$LOLIM
        030312  177777                  .WORD   T$HILIM
    274 030314                          GPRMD   DREMSG,DET,D,177777,0,177777,YES
        030314  025052                  .WORD   T$CODE
        030316  031106                  .WORD   DREMSG
        030320  177777                  .WORD   177777
        030322  000000                  .WORD   T$LOLIM
        030324  177777                  .WORD   T$HILIM
    275 030326                  3$:     GPRML   STLMT,OPFLG,1,YES
        030326  024130                  .WORD   T$CODE
        030330  031434                  .WORD   STLMT
        030332  000001                  .WORD   1
    276 030334                          XFERF   2$
        030334  013044                  .WORD   T$CODE
    277 030336                          GPRMD   DAMSG,DAT,D,177777,1,177776,YES
        030336  003052                  .WORD   T$CODE
        030340  031121                  .WORD   DAMSG
        030342  177777                  .WORD   177777
        030344  000001                  .WORD   T$LOLIM
        030346  177776                  .WORD   T$HILIM
    278 030350                          GPRMD   SKMSG,SKT,D,177777,1,177776,YES
        030350  004052                  .WORD   T$CODE
        030352  031151                  .WORD   SKMSG
        030354  177777                  .WORD   177777
        030356  000001                  .WORD   T$LOLIM
        030360  177776                  .WORD   T$HILIM
    279 030362                  2$:     GPRML   CHANGE,CHFLG,1,YES
        030362  010130                  .WORD   T$CODE
        030364  031221                  .WORD   CHANGE
        030366  000001                  .WORD   1
```

```
280 030370
    030370  107044                      XFERF   1$
                                        .WORD   T$CODE
281 030372
    030372  034130                      GPRML   STIPMS,STIP,1,YES
    030374  030654                      .WORD   T$CODE
    030376  000001                      .WORD   STIPMS
                                        .WORD   1
282 030400
    030400  013044                      XFERF   6$
                                        .WORD   T$CODE
283 030402
    030402  011052                      GPRMD   MXBUF,MXB,D,177777,3,5120.,YES
    030404  031255                      .WORD   T$CODE
    030406  177777                      .WORD   MXBUF
    030410  000003                      .WORD   177777
    030412  012000                      .WORD   T$LOLIM
                                        .WORD   T$HILIM
284 030414
    030414  022052                      GPRMD   MINBUF,MNB,D,177777,3.,5120.,YES
    030416  031266                      .WORD   T$CODE
    030420  177777                      .WORD   MINBUF
    030422  000003                      .WORD   177777
    030424  012000                      .WORD   T$LOLIM
                                        .WORD   T$HILIM
285 030426                      6$:     GPRML   RDONLY,ROF,1,YES
    030426  026130                      .WORD   T$CODE
    030430  030743                      .WORD   RDONLY
    030432  000001                      .WORD   1
286 030434                              GPRML   RANPAT,RAN,1,YES
    030434  027130                      .WORD   T$CODE
    030436  030753                      .WORD   RANPAT
    030440  000001                      .WORD   1
287 030442                              XFERT   7$
    030442  006024                      .WORD   T$CODE
288 030444                              GPRMD   ONLONE,PAT,0,17,0,7,YES
    030444  030032                      .WORD   T$CODE
    030446  030763                      .WORD   ONLONE
    030450  000017                      .WORD   17
    030452  000000                      .WORD   T$LOLIM
    030454  000007                      .WORD   T$HILIM
289 030456                      7$:     GPRMD   CMMSG,RDT,D,177777,0,128.,YES
    030456  006052                      .WORD   T$CODE
    030460  031553                      .WORD   CMMSG
    030462  177777                      .WORD   177777
    030464  000000                      .WORD   T$LOLIM
    030466  000200                      .WORD   T$HILIM
290 030470                              GPRMD   DEMSG,DDT,D,177777,0,175,YES
    030470  007052                      .WORD   T$CODE
    030472  031025                      .WORD   DEMSG
    030474  177777                      .WORD   177777
    030476  000000                      .WORD   T$LOLIM
    030500  000175                      .WORD   T$HILIM
291 030502                              GPRMD   MXHD,MXH,D,100,0,1,YES
    030502  012052                      .WORD   T$CODE
    030504  031277                      .WORD   MXHD
    030506  000100                      .WORD   100
    030510  000000                      .WORD   T$LOLIM
    030512  000001                      .WORD   T$HILIM
292 030514                              GPRMD   MINHD,MNH,D,100,0,1,YES
    030514  013052                      .WORD   T$CODE
    030516  031306                      .WORD   MINHD
```

```
                030520  000100                          .WORD   100
                030522  000000                          .WORD   T$LOLIM
                030524  000001                          .WORD   T$HILIM
        293     030526                                  GPRML   ASK,ANS,1,YES
                030526  037130                          .WORD   T$CODE
                030530  030614                          .WORD   ASK
                030532  000001                          .WORD   1
        294     030534                                  XFERF   15$
                030534  013044                          .WORD   T$CODE
        295     030536                                  GPRMD   MXCYL,MXC,D,177600,0,511.,YES
                030536  014052                          .WORD   T$CODE
                030540  031315                          .WORD   MXCYL
                030542  177600                          .WORD   177600
                030544  000000                          .WORD   T$LOLIM
                030546  000777                          .WORD   T$HILIM
        296     030550                                  GPRMD   MINCYL,MNC,D,177600,0,511.,YES
                030550  015052                          .WORD   T$CODE
                030552  031325                          .WORD   MINCYL
                030554  177600                          .WORD   177600
                030556  000000                          .WORD   T$LOLIM
                030560  000777                          .WORD   T$HILIM
        297     030562                          15$:    GPRMD   MXSEC,MXS,D,77,0,39.,YES
                030562  016052                          .WORD   T$CODE
                030564  031335                          .WORD   MXSEC
                030566  000077                          .WORD   77
                030570  000000                          .WORD   T$LOLIM
                030572  000047                          .WORD   T$HILIM
        298     030574                                  GPRMD   MINSEC,MNS,D,77,0,39.,YES
                030574  017052                          .WORD   T$CODE
                030576  031356                          .WORD   MINSEC
                030600  000077                          .WORD   77
                030602  000000                          .WORD   T$LOLIM
                030604  000047                          .WORD   T$HILIM
        299
        300     030606                          1$:     GPRML   AUTOMS,AUTO,1,YES
                030606  033130                          .WORD   T$CODE
                030610  031377                          .WORD   AUTOMS
                030612  000001                          .WORD   1
        301
        302     030614                                  ENDSFT
                                                        .EVEN
                030614                          L10025:
        303
        304
        308
        309     030614  103     110     101     ASK:    .ASCIZ  /CHANGE VALUES OF MXCYL & MINCYL/
        310     030654  123     124     111     STIPMS: .ASCIZ  %STIPULATE R/W XFER SIZE%
        311     030704  123     105     105     SRTMSG: .ASCIZ  /SEEK RETRY LMT/
        312     030723  043     040     117     CHKLMT: .ASCIZ  /# OF ERR DUMPED/
        313     030743  122     104     040     RDONLY: .ASCIZ  /RD ONLY/
        314     030753  122     101     116     RANPAT: .ASCIZ  /RAN PAT/
        315     030763  127     110     111     ONLONE: .ASCIZ  /WHICH ONE/
        316     030775  110     122     104     ERMSG:  .ASCIZ  /HRD ERR LMT/
        317     031011  123     106     124     SFTMSG: .ASCIZ  /SFT ERR LMT/
        318     031025  043     040     117     DEMSG:  .ASCIZ  /# OF DATA ERR RPT'D PER BUF/
        319     031061  122     105     124     RTMSG:  .ASCIZ  /RETRY LMT/
        320     031073  123     113     040     SEMSG:  .ASCIZ  /SK ERR LMT/
```

```
 321 031106      104     122     040  DREMSG: .ASCIZ  /DR ERR LMT/
 322 031121      104     101     124  DAMSG:  .ASCIZ  /DATA XFER LMT (*10(10))/
 323 031151      123     113     040  SKMSG:  .ASCIZ  /SK LMT (*10(3))/
 324 031171      124     111     115  INMSG:  .ASCIZ  /TIME BETW REPORTS (MIN)/
 325 031221      103     110     101  CHANGE: .ASCIZ  %CHANGE SEEK, R/W PARAMETERS%
 326 031255      115     101     130  MXBUF:  .ASCIZ  /MAX XFER/
 327 031266      115     111     116  MINBUF: .ASCIZ  /MIN XFER/
 328 031277      115     101     130  MXHD:   .ASCIZ  /MAX HD/
 329 031306      115     111     116  MINHD:  .ASCIZ  /MIN HD/
 330 031315      115     101     130  MXCYL:  .ASCIZ  /MAX CYL/
 331 031325      115     111     116  MINCYL: .ASCIZ  /MIN CYL/
 332 031335      123     124     101  MXSEC:  .ASCIZ  /STARTING MAX SEC/
 333 031356      123     124     101  MINSEC: .ASCIZ  /STARTING MIN SEC/
 334 031377      103     110     113  AUTOMS: .ASCIZ  /CHK DRDY/
 335 031410      104     101     124  FDCHK:  .ASCIZ  /DATA DMP ON DCK ERR/
 336 031434      104     122     117  STLMT:  .ASCIZ  /DROP DR ON OPER LMTS REACHED/
 337 031471      104     122     117  DRPMS:  .ASCIZ  /DROP DR ON ERR LMTS REACHED/
 338 031525      104     101     124  DERPMS: .ASCIZ  /DATA MISCOMPARE LIMIT/
 339 031553      127     117     122  CMMSG:  .ASCIZ  /WORDS PER SECTOR COMPARED ON READ/
 340
 341                                   .EVEN
 342
 346
 347
 348 031616                                    ENDMOD
 349
 350             032114                .=32114          ;THIS CORRECTS FOR 'APT' MAILBOX
 351 032114                            LASTAD
                                               .EVEN
     032114                            L$LAST::
```

```
      1                                 .SBTTL  DIAGNOSTIC SUPERVISOR -- LOW CORE SET UP
```

```
     35 062710  000000                    .WORD   0               ;SPACE FOR USER POOL POINTER
     36 062712  000000                    .WORD   0               ;SIZE
     37 062714  000000                    .WORD   0               ;CHECKSUM (NOT CURRENTLY USED)
     38 062716  000000                    .WORD   0               ;SIZE OF H.W. PTAB. ALLOCATION
     39         062722                    END.SUPV=.+2
     40         000200                    .END 200
```

```
ABOFLA  032440 G    BIT7  = 000200 G    CLKSON  032412 G    C$ESCA= 000010    DEV.CO  032126 G
ABOPAS  032356 G    BIT8  = 000400 G    CLK.SE  037162       C$ESEG= 000005    DIAGMC= 000000
ABO.FM  034720      BIT9  = 001000 G    CLMT  = 000064       C$ESUB= 000003    DIAG.T  032446 G
ADDCOD  012060 G    BLD.HW  037602      CLNCOD  011714 G    C$ETST= 000001    DIFMSG  002465
AFREAD  017006      BLOCK   055214      CLRDAT  010730      C$EXIT= 000032    DIFWD = 000066
AFSI    032146 G    BMP   = 000042      CLR.MA  037436      C$GMAN= 000043    DLT   = 010000
AFWRCK  017026      BPRIOR  002212      CMMSG   031553      C$GPHR= 000042    DLTCNT= 000026
AGSTAT  017250      BRMSG   030101      CMRD    010262      C$GPRI= 000040    DMPBUF  024354
ALLOC   053060      BSECPT= 000112      CMSK    002154      C$GTIM= 000052    DMPDCK  003122
ANS   = 000076      BSEC0   025446      CNT   = 000012      C$INIT= 000011    DNRDY   002526
APT.ER  034050      BSEC1   025510      CNTFLG  002334      C$INLP= 000020    DOWCK = 000076
ARDHDR  017106      BSEC2   025552      CNTLR1  002176      C$KWOF= 000035    DPDVD   062056 G
ASEEK   016744      BSEC3   025614      CNTLR2  002200      C$KWON= 000034    DPHOUR= 000070
ASK     030614      BSEC4   025656      CNTYPE  030060      C$LOOP= 000100    DPMIN = 000071
ASSEMB= 000011      BSEC5   025720      CNVT    055660      C$MANI= 000051    DPMUL   061744 G
AUTO  = 000066      BSEC6   025762      COMMAN  032164 G    C$MSG = 000023    DRBT  = 000010
AUTOMS  031377      BSEC7   026024      COMMTA  055474      C$PNTB= 000014    DRBUF   026514
AWRITE  017354      BUF1    002316      CONTCL  061272 G    C$PNTF= 000017    DRDRV   021502
A$AAV   036716      BUF2    002320      CONWR   022652      C$PNTS= 000016    DRDY  = 000001
A$AAW   036732      BUSAD   003770      CRDY  = 000200      C$PNTX= 000015    DREMSG  031106
A$AAX   036744      BVEC    002210      CRLBA   002427      C$POIN= 000040    DRFLG = 000042
A$AAY   036752      B$AAB   041204      CRLCS   002377      C$QIO = 000377    DRLMT   010320
A$AAZ   036766      B$AAF   041116      CRLDA   002441      C$RDBU= 000007    DRMSG   030143
A$ABA   036776      CALLPC= 000022      CRLF    051572      C$REFG= 000050    DRNM    003666
BA    = 000002      CALLPS= 000024      CRLMP   002453      C$REQT= 000045    DROP    004134
BA16  = 000020      CALLSP= 000026      CS    = 000000      C$RESE= 000033    DROPCO  012144 G
BA17  = 000040      CALLTC= 000030      CSR   = 000000      C$REVI= 000002    DRPMS   031471
BBA   = 000110      CAL.CL  057602      CSRMSG  030065      C$RPT = 000025    DRPRS   002137
BCSADR= 000046      CAL.TI  057640 G    CSTUFF  022420      C$SEFG= 000047    DRSEL = 000106
BCSR    002206      CART    002476      CURR.S  032122 G    C$SPRI= 000041    DRST  = 000013
BDA   = 000040      CEND    022276      CURR.T  032124 G    C$SVEC= 000037    DRTYPE  030112
BDRSEL  002214      CHANGE  031221      CYL     002164      C$TPRI= 000013    DRUT    002136
BGN.SU= 032114      CHFLG = 000020      CYLMSK  002150      C$UNBU= 000031    DRVER   002716
BINMSG  051370      CHKFNC  016652      C$AAD   044462      C$WTM = 000026    DSE   = 000400
BIT0  = 000001 G    CHKLMT  030723      C$AAE   044474      C$WTU = 000027    DSPCOD  010346 G
BIT00 = 000001 G    CHKLUP  041220      C$AAK   045472      C.HDR   002312    DUNIT.  032362 G
BIT01 = 000002 G    CHKSEC  002220      C$AAL   045636      DA    = 000004    DVC.FT  045442
BIT02 = 000004 G    CHKSTR  053422      C$ABRT= 000021      DALMT   010254    DWCNT   002342
BIT03 = 000010 G    CHKTTY  051510      C$ADR = 000020      DAMSG   031121    DWCNT1  002344
BIT04 = 000020 G    CHK.MA  037360      C$AU  = 000054      DAT   = 000006    D$AAG   046346
BIT05 = 000040 G    CHK.PC  044510      C$BRK = 000022      DATCER= 000074    D$AAH   046364
BIT06 = 000100 G    CHK.SW  033550      C$BSEG= 000004      DCD   = 000074    D$AAI   051132
BIT07 = 000200 G    CHRCNT  052742      C$BSUB= 000002      DCDMSG  003244    D$AAJ   051136
BIT08 = 000400 G    CH.FLA  037066      C$BUFF= 000030      DCKFG = 000040    D$AAK   051154
BIT09 = 001000 G    CH.PAS  037104      C$CEFG= 000046      DCRC  = 004000    D$AAL   051172
BIT1  = 000002 G    CKBDSC  025300      C$CLEA= 000012      DCRCER= 000022    D$AAM   051202
BIT10 = 002000 G    CKBDTK  025356      C$CLP1= 000006      DCS   = 000104    EF.CON= 000036 G
BIT11 = 004000 G    CKDATA  021724      C$CVEC= 000036      DDT   = 000016    EF.NEW= 000035 G
BIT12 = 010000 G    CKDERR  017636      C$DCLN= 000044      DECMSG  051404    EF.PWR= 000034 G
BIT13 = 020000 G    CLEAR   025256      C$DODU= 000053      DECNT   002222    EF.RES= 000037 G
BIT14 = 040000 G    CLEAR.  040502      C$DRPT= 000024      DELMT   010264    EF.STA= 000040 G
BIT15 = 100000 G    CLKACC  032354 G    C$DU  = 000055      DEMSG   031025    EF01  = 000001 G
BIT2  = 000004 G    CLKBFR  057604      C$EDIT= 000002      DERCNT= 000020    EF02  = 000002 G
BIT3  = 000010 G    CLKCNT  032352 G    C$ERDF= 000002      DERMSG  003267    EF03  = 000003 G
BIT4  = 000020 G    CLKJUM  060210 G    C$ERHR= 000003      DERPMS  031525    EF04  = 000004 G
BIT5  = 000040 G    CLKRES  061212 G    C$ERSF= 000001      DERR  = 040000    EF05  = 000005 G
BIT6  = 000100 G    CLKSER  061346 G    C$ERSO= 000004      DET   = 000052    EF06  = 000006 G
```

```
EF07  = 000007 G    E.DA    002302      FMT9A   006651      G$RADD= 000040      I$MSG  = 000041
EF08  = 000010 G    E.MP    002304      FORM.T  046010      G$RADF= 000200      I$PWR  = 000041
EF09  = 000011 G    E.MP1   002306      FREE    053316      G$RADL= 000120      I$RPT  = 000041
EF10  = 000012 G    E.MP2   002310      FRMT16  007273      G$RADO= 000020      I$SEG  = 000041
EF11  = 000013 G    FASCII  002336      FUNC  = 000044      G$RADT= 000100      I$SFT  = 000041
EF12  = 000014 G    FASPNT  002340      F$AU  = 000015      G$XFER= 000004      I$SRV  = 000041
EF13  = 000015 G    FDCHK   031410      F$BGN = 000040      G$YES = 000010      I$SUB  = 000041
EF14  = 000016 G    FILINF  012206      F$CLEA= 000007      HCE   = 040000      I$TST  = 000041
EF15  = 000017 G    FILL    052240      F$DU  = 000016      HCORED  036656      J$JMP  = 000167
EF16  = 000020 G    FILL.C  000204 G    F$END = 000041      HCOREQ  036566      KBPTR   032224 G
ELT   = 000002      FILTD   012232      F$HARD= 000004      HCORET  032402 G    KBUF    032226 G
EMT.TR  032444 G    FINDBF  011564      F$HW  = 000013      HCRC  = 004000      KILLDC  002174
END     011236      FINERR  017600      F$INIT= 000006      HCRCER= 000024      LDFUNC  015226
ENDBUF  027774      FLAGS   032160 G    F$JMP = 000050      HC.ADR  032152 G    LIMIT   010246
ENDOFP  026506      FLAGS1  032162 G    F$MOD = 000000      HC.DEF  032144 G    LINE.F  032442 G
ENDWR   022714      FLAGTA  055412      F$MSG = 000011      HC.DIA  032142 G    LINE1   005402
END.OF  040470      FLAG.I  037146      F$PWR = 000017      HDHOME  023752       LINE2   005606
END.SU= 062722      FLA.SE  055360      F$RPT = 000012      HDRFND  002216       LINE3   005716
ENVIRO  032166 G    FLG.MA  037106      F$SEG = 000003      HEAD  = 000100      LIST    020620
EOP.CH  061370 G    FMTDT   007552      F$SOFT= 000005      HERTZ.  036526       LOAD.F  037102
EOP.FM  034734      FMTS1   007401      F$SRV = 000010      HINUM   002144       LOGMSG  051412
EOP.IN  037100      FMTS1A  007464      F$SUB = 000002      HNF   = 010000      LONUM   002146
EPS     003712      FMTS1B  007504      F$SW  = 000014      HNFERR= 000032      LPBFR   032222 G
ERLMT   010250      FMTS2   007537      F$TEST= 000001      HOLDSP= 000020      LPCNTR  032220 G
ERLMTM  003157      FMTS2A  007603      GARBAG  052744      HOUR    002274       LPS     003677
ERMSG   030775      FMTS2B  007672      GDDAT   002260      HPTCOD  010226 G    LPT.AD  036544
ERR   = 100000      FMTS3   007727      GETCHR  051450      HRDPRM  027776 G    LPT.RE  036540
ERRCNT= 000012      FMTS3A  010036      GETCMN  055034      HWSEC   003556       LSI.RE  036534
ERREX   017476      FMTS4   010073      GETDST  022372      HW.ADR  032150 G    LSTDA = 000064
ERRFOR  045714      FMTS5   010166      GETFNC  013726      H$AAB   056206       LSTDR1  002202
ERRHAN  044514      FMTXS   007342      GETPAR  046526      ILLEG   003626       LSTDR2  002204
ERRHDR  004334      FMT1    006236      GETSWI  054030      INBAD   023510       LSTHDR= 000050
ERRVEC  002346      FMT1A   006273      GETSYS  023630      INCALL  002356       LSTTIM  002266
ERR.HR  045452      FMT10   006725      GET.TW  053600      ININIT  032372 G    LUP     057506
ERR.NU  032116 G    FMT10A  006761      GHDR    020300      INITCO  010450 G    LUP.AD  044512
ERR.SF  045456      FMT10B  007032      GLBDAT  002126 G    INITIA  051420       L$APT   002036 G
ERR1    004456 G    FMT11   007040      GLBEQA  002126 G    INIT.M  037504       L$AU    012060 G
ERR1FO  046000      FMT12   007060      GLBERR  004456 G    INIT.R  032206 G    L$AUT   002074 G
ERR10   005234 G    FMT13   007070      GLBSUB  012206 G    INMSG   031171       L$CCP   002106 G
ERR12   005304 G    FMT13D  007105      GLBTXT  002360 G    INPUTA  052346       L$CLEA  011714 G
ERR13   005312 G    FMT14   007132      GOERRX  016642      INTEN = 000100      L$CO    002032 G
ERR2    004464 G    FMT14A  007141      GOFIN   016646      INTERV  002264       L$DEPO  002011 G
ERR3    004550 G    FMT14B  007153      GSBIT = 000003      INTFOR  045644       L$DESC  002102 G
ERR4    004700 G    FMT14C  007150      GSTAT = 000004      INTR1   015336 G    L$DEVP  002064 G
ERR6    004752 G    FMT15   007215      GSTFNC  014400      INTR2   015346       L$DISP  010350 G
ERR7    005012 G    FMT17   006260      GWCDA   024046      INVAL.  036452       L$DR    002112 G
ERR8    005050 G    FMT18   007335      G$EXCP= 000400      INVINT  045502       L$DRCT  002070 G
ERR9    005170 G    FMT2    006302      G$HILI= 000002      INV.SW  033504       L$DRS   002072 G
ERT     004017      FMT2A   006323      G$LOLI= 000001      IN.SUF  040454       L$DRST  002112 G
ESC.PC  044506      FMT3    006345      G$NO  = 000000      ISDRST  022406       L$DTP   002040 G
EV.COU  032120 G    FMT3A   006406      G$OFFS= 000400      ISSUE   015174       L$DU    012144 G
EXHAUS  002630      FMT4    006472      G$OFSI= 000376      I$AU  = 000041       L$DUT   002076 G
EXIT    017440      FMT5    006525      G$PRMA= 000001      I$CLN = 000041       L$DVTY  002114 G
EXIT1   017472      FMT6    006541      G$PRMD= 000002      I$DU  = 000041       L$EF    002056 G
EXP     004112      FMT7    006567      G$PRML= 000000      I$HRD = 000041       L$EFLG  002034 G
E.BA    002300      FMT8    006603      G$RADA= 000140      I$INIT= 000041       L$EXP1  002042 G
E.CS    002276      FMT9    006613      G$RADB= 000000      I$MOD = 000041       L$EXP2  002044 G
```

```
L$EXP3  002046 G    MBDMSC  003525       MXHD    031277       POWER   011712       RPTCOD  010352 G
L$HARD  030000 G    MDCER   003072       MXS   = 000034       PRFLGS= 000056       RSEEK = 000114
L$HPCP  002016 G    MDERS   002725       MXSEC   031335       PRGER   002567       RSTACK  061540 G
L$HPTP  002022 G    MDHEDR  002000 G     NEWPRI  061336 G     PRINTC  052720       RTMSG   031061
L$HW    010230 G    MDSER   002753       NEXTAR  055576       PRINTF  056226       RTYPE = 000052
L$ICP   002104 G    MEM.SI  036554       NOCRDY  002516       PRIOR = 000004       RT1     004067
L$INIT  010430 G    MFUNC   002411       NODRIV  003654       PRIOR1  002254       RWCNT   002130
L$LADP  002026 G    MHDER   003107       NOLOAD  003430       PRIOR2  002256       RXFR1 = 000002
L$LAST  032114 G    MINBUF  031266       NOPWR   003737       PRI00 = 000000 G     RXFR2 = 000004
L$MREV  002050 G    MINCYL  031325       NORDDC  004254       PRI01 = 000040 G     RXFR3 = 000060
L$NAME  002000 G    MINHD   031306       NORDY   002541       PRI02 = 000100 G     SAVE    015354
L$REPP  002066 G    MINSEC  031356       NOREV   003466       PRI03 = 000140 G     SAVEDO= 034050
L$REV   002010 G    MINUTE  002272       NO.CLK  036502       PRI04 = 000200 G     SEARCH  053546
L$RPT   010352 G    MIN.IN  032172 G     NO.FLA  055372       PRI05 = 000240 G     SEC     002170
L$SOFT  030154 G    MIN.US  032174 G     NO.LPT  052710       PRI06 = 000300 G     SECMSK  002152
L$SPC   002062 G    MK    = 000001       NO.PTA  036706       PRI07 = 000340 G     SECOND  002270
L$SPCP  002020 G    MNB   = 000044       NR    = 000000       PRNTST  052610       SEEK  = 000006
L$SPTP  002024 G    MNC   = 000032       NUMBIN  046034       PRO.CM  037060       SEGSTA  032414 G
L$STA   002030 G    MNH   = 000026       NUM.LA  046202       PRPOS = 000124       SEL   = 000046
L$SW    010246 G    MNS   = 000036       NUM.NO  032156 G     PTAB.S  032400 G     SELMT   010252
L$TIML  002014 G    MODR    061656 G     NUM.UN  032564       PUTCHR  051424       SEMSG   031073
L$TIMU  002054 G    MP    = 000006       NUNITS  041172       PWRFLG  002326       SERLMT  003201
L$TIM1  002052 G    MPT     003614       NWRTS   002573       PWR.FA  062550 G     SERNM1= 000100
L$TSTI  002100 G    MRDER   003047       NXM   = 020000       PWR.FL  032204 G     SERNM2= 000102
L$UNIT  002012 G    MRLCS   002367       NXMCNT= 000034       PWR.MS  062676       SET   = 000004
L.CLK.  036512      MRT     004006       NXTFOR  055652       PWR.SA  062672       SET.MA  037272
L10000  004462      MSFER   002677       OCTMSG  051376       PWR.UP  062674       SEXHAU  003360
L10001  004546      MSG.AD  032140 G     ODRDRV  021476       P.CLK.  036520       SFEMSG  003222
L10002  004676      MSG.TY  032114 G     ONLONE  030763       RAN   = 000056       SFLMT   010314
L10003  004750      MSKER   002666       OPCALL  002354       RAND    022464       SFTCNT= 000014
L10004  005010      MST     004035       OPFLG = 000050       RANPAT  030753       SFTMSG  031011
L10005  005046      MSTART  004216       OPI   = 002000       RCD     004123       SFTPRM  030152 G
L10006  005166      MST1    004052       OPICNT= 000030       RCNT    020600       SHIFT   062376 G
L10007  005232      MSWRPK  004236       OPROK   014166       RDBDSC  020636       SIGN  = 000004
L10010  005302      MTCR    004366       OVER    003310       RDDFNC  015132       SKCNT = 000000
L10011  005310      MTDCRC  004165       O$APTS= 000001       RDHDR = 000010       SKCNT1= 000054
L10012  005400      MTDLT   004172       O$AU  = 000001       RDHFNC  015050       SKDON = 000001
L10013  010244      MTDRV   004211       O$BGNR= 000001       RDNHC   014336       SKECNT= 000016
L10014  010346      MTEST   013152       O$BGNS= 000001       RDONLY  030743       SKFNC   014420
L10015  010426      MTGS    004376       O$DU  = 000001       RDT   = 000014       SKHS  = 000020
L10016  011712      MTHCRC  004157       O$GNSW= 000001       READ  = 000014       SKLMT   010256
L10017  012056      MTHNF   004152       O$POIN= 000001       READ.P  057610 G     SKMSG   031151
L10020  012142      MTNXM   004204       O$PWR = 000001       RECNT   002126       SKRD    014076
L10021  012204      MTOPI   004177       PARSES  055106       REGBAC  062300 G     SKRDRD  014126
L10022  026510      MTRD    004436       PAR.LA  051074       REGEN   002172       SKRETR= 017166
L10023  017576      MTRH    004416       PASS.C  032130 G     REGSAV  062264 G     SKRH    014246
L10024  030060      MTRNH   004446       PAT   = 000060       REPORT  012442       SKT   = 000010
L10025  030614      MTSK    004406       PATLST  026066       REQ     003345       SKTO  = 010000
MAIN    013406      MTWR    004426       PAT0    026106       REQN.P  032170 G     SKWRT   014036
MAJ.IN  032176 G    MUL     061612 G     PAT1    026146       REQN.T  037062       SMSG    002616
MAJ.LO  057606      MVCER   002767       PAT2    026206       RESTAR  011324       SMSK    002156
MAJ.US  032200 G    MVEC    003642       PAT3    026246       RETRY = 000036       SOFTCS= 000116
MAN.TI  001244      MXB   = 000022       PAT4    026306       RE.SET  033652       SOPLMT  003446
MAP16   062314 G    MXBUF   031255       PAT5    026346       RLT   = 000000       SPE   = 004000
MASK.B  041216      MXC   = 000030       PAT6    026406       RNTEMP  002262       SPEC.U  037006
MASK.W  041214      MXCYL   031315       PAT7    026446       ROF   = 000054       SPTCOD  010244 G
MAXWC   002322      MXH   = 000024       PEROTH  013416       RPS     003724       SPV.SE  000400
```

```
SRLT  = 000062        TEMP0   002224        T$CODE= 033130        T.MNB   010312        WRBUF   020352
SRTMSG  030704        TEMP1   002226        T$ERCO= 000063        T.MNC   010300        WRCHK = 000002
STARTC  061266 G      TEMP2   002230        T$ERRN= 000307        T.MNH   010274        WRINIT  002160
STDMP   025070        TEMP3   002232        T$EXCP= 000000        T.MNS   010304        WRIPG = 000122
STFLG   002332        TEMP4   002234        T$HILI= 000047        T.MXB   010270        WRITE = 000012
STIP  = 000070        TEMP5   002236        T$LOLI= 000000        T.MXC   010276        WRPACK  022562
STIPMS  030654        TEMP6   002240        T$LSYM= 010000        T.MXH   010272        WRPOS   002162
STLMT   031434        TEMP7   002242        T$NEST= 177777        T.MXS   010302        WRTCKF  014356
STRCHR  052300        TEMP8   002244        T$NSK0= 000000        T.PAT   010326        WRTFNC  015062
STRT.T  037064        TEMP9   002246        T$NSK1= 000005        T.RAN   010324        WTRDY   022322
STWRT   022736        TERMI   057576        T$SAVL= 177777        T.ROF   010322        WXFR1 = 000006
ST.SET  033716        TERMLI  055400        T$SEGL= 177777        T.SLT   010330        WXFR2 = 000010
ST1     002350        TERMTA  051362        T$SUBN= 000000        T.STA   010316        WXFR3 = 000062
ST2     002352        TEST.M  037020        T$TAGL= 177777        T.STIP  010336        XCHFLG  010266
SUNIT.  037070        TIME    002360        T$TAGN= 010026        T.WCK   010340        XEQDIA  061424 G
SUPERV  034752        TIMFLG  032350 G      T$TEMP= 000000        T1      013152 G      XEQSUB  061412 G
SUPFLA  032360 G      TIM.CO  032202 G      T$TEST= 000001        UDERR   002644        XEQ.CL  041134
SUPV.T  032532 G      TIM.OP  046006        T$TSTM= 177777        UNIT.D  032132 G      XEQ.CM  036444
SUP.PR  033470        TOO.MA  051342        T$TSTS= 000001        UNI.MA  037010        XEQ.IN  040616
SUR     002166        TRACK   003142        T$$AU = 010020        UNLOAD  003404        XEQ.LA  034706
SVCGBL= 000000        TRERR = 000072        T$$CLE= 010017        USER.P  032374 G      XEQ.OP  040710
SVCHAN  041406        TRPFLG  002330        T$$DU = 010021        USER.T  032376 G      XEQ.PR  034110
SVCINS= 000000        TRPHAN  012434        T$$HAR= 010024        UUT     002324        XEQ.TE  040754
SVCSUB= 177777        TSTDRV  002134        T$$HW = 010013        VALID.  032634        XEXIT   016740
SVCTAG= 000000        TST.AB  041330        T$$INI= 010016        VAL.LA  033454        XTIME   060276 G
SVCTST= 177777        TST.TO  033532        T$$MSG= 010012        VAL.SW  037120        XTIMEN  061122
SWCHAN  036700        TYINT   010260        T$$RPT= 010015        VC    = 001000        XTIMST  060320
SWITCH  055552        TYPDR = 000006        T$$SOF= 010025        VECMSG  030134        XXDP.D  036464
SWSEC   003576        TYPEC   051736        T$$SRV= 010023        VECT  = 000002        X$ALWA= 000000
SW.ADR  032154 G      TYPEPC  045632        T$$SW = 010014        VECT1   002250        X$FALS= 000040
SW.PTA  036664        TYPFLA  055254        T$$TES= 010022        VECT2   002252        X$OFFS= 000400
SYSCLK  002314        TYPLIN  051634        T.ANS   010344        WCK   = 000072        X$TRUE= 000020
SYSMSK  002142        TYPNUM  051216        T.AUT   010334        WDE   = 100000        $BREG   037160
SYS.FT  045432        TYPSTR  051654        T.CLT   010332        WGE   = 002000        $ENDAD  061376 G
S$LSYM= 010000        TYP.ER  045462        T.DCD   010342        WGEST   003014        $SAV2   062442 G
TAGX    013564        TYT   = 000012        T.DCK   010306        WHY     002132        $SAV3   062456 G
TDR   = 000120        TY.UNI  040474        T.DRIV  002140        WIDTH   046402        $SAV4   062474 G
TELCYL  006076        T$ARGC= 000001        T.DRP   010310        WL    = 020000        $SAV5   062514 G

. ABS.  062720     000
        000000     001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  19984 WORDS  ( 79 PAGES)
DYNAMIC MEMORY AVAILABLE FOR  70 PAGES
CZRLKA.BIN,CZRLKA=#SVCRT/M,CZRLKA,DOCTOR
```