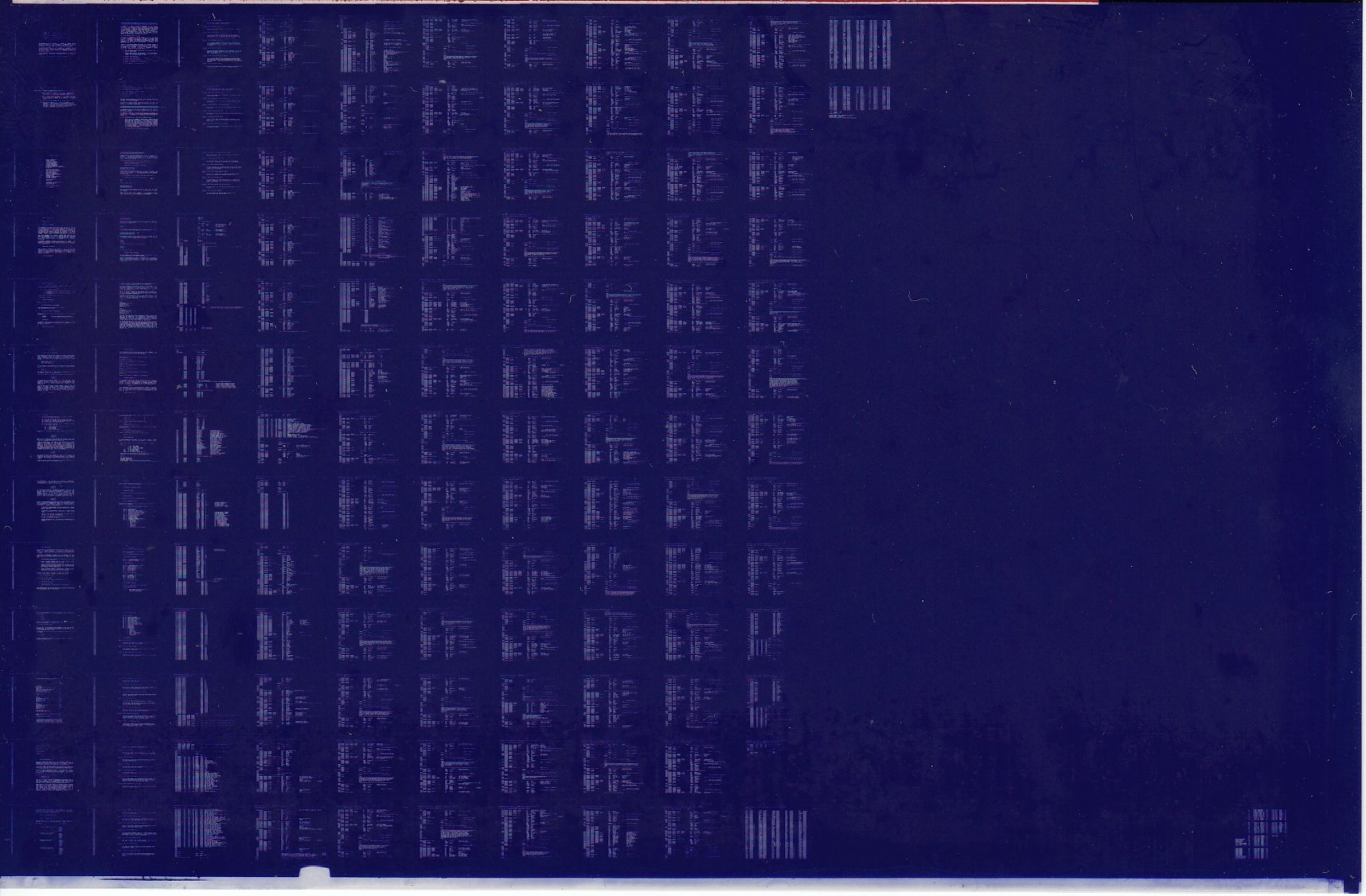
11/Z1+ RLV11 RL11/RLV11 CTLR TST 2
RL01/02 CNRLHA0

AH-T758A-MC
FICHE 1 OF 1

APR 1984
digital
Made In USA



```
                        .TITLE  CNRLHAO RL11/RLV11 CTLR TST 2
                        .NLIST TOC
                        .REM @
```

IDENTIFICATION

PRODUCT CODE: AC-T757A-MC

PRODUCT NAME: CNRLHAO RL11/RLV11 CONTROLLER TEST 2

PRODUCT DATE: DECEMBER 19, 1983

MAINTAINER: ISS DIAGNOSTIC SERVICES

AUTHOR: JAMES S. DOUCETTE

```
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
```

REVISION HISTORY

CHANGES MADE TO CZRLHBO IN PRODUCING CNRLHAO FOR THE SBC-11/21+ (FALCON-PLUS), DEC. 19, 1983. CHANGES ARE IDENTIFIED BY ";JSD REV A".

1. CHANGED THE FORM OF THE ARGUMENT TO ALL "DELAY" MACRO CALLS FROM #<VALUE> TO <VALUE>. THE FORMER GAVE ASSEMBLY ERRORS UNDER THE VAX/VMS DEVELOPMENT ENVIRONMENT (MCR MAC).

2. CHANGED THE GENERAL OPERATING PRIORITY OF THE PROGRAM FROM LEVEL 7 TO LEVEL 6 TO ALLOW THE "BREAK" KEY TO INVOKE ODT. (THE TRAP HANDLER AND DEVICE INTERRUPT SERVICE ROUTINES STILL RUN BRIEFLY AT LEVEL 7).

3. SET VECTOR 140 WITH THE ADDRESS OF ODT IN ROM (170000).

4. PREVENTED THE EXECUTION OF TEST 7, WHICH IS DEPENDENT ON CLOCK INTERRUPTS. UNDER FALCON-PLUS, CLOCK OPERATION IS NOT GUARANTEED. CLOCK INTERRUPTS MAY OR MAY NOT BE HARD-ENABLED, AND EVEN IF THEY WERE, THE INTERRUPT RATE COULD BE 50, 60, OR 800 HERTZ. FURTHERMORE, THE DRS CLOCK MACROS RETURN MISLEADING INFORMATION (UNDER FALCON-PLUS).

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

1.1.1 STRUCTURE OF PROGRAM

THIS DIAGNOSTIC IS COMPATIBLE WITH BOTH CNDP+ AND ACT. IT CAN BE RUN STANDALONE UNDER CNDP+, AND CAN BE CHAINED UNDER CNDP+, ACT AND APT IN ACT MODE (SEE 2.2 "CHAIN MODE OPERATION" FOR DETAILS OF CHAINING PROCEDURE). IT IS A SINGLE PROGRAM FROM THE STANDPOINT OF THE DIAGNOSTIC USER, WHICH AT RUN TIME IS APPENDED TO A COMMON FRONT-END PIECE OF SUPERVISOR SOFTWARE THROUGH WHICH THE DIAGNOSTIC PROGRAM INTERFACES TO THE ENVIRONMENT AS IT EXECUTES. (IN THIS DOCUMENT, "CNDP+" REFERS TO THE FALCON-SPECIFIC XXDP+ SYSTEM).

WHEN THIS DIAGNOSTIC IS STARTED, 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 (DR>). AT COMMAND MODE THE OPERATOR MAY ENTER ANY OF SEVERAL COMMANDS AS DESCRIBED IN 2.0 "OPERATING INSTRUCTIONS".

THE DIAGNOSTIC PROGRAM IS LOADED IN THE LOWER 8K OF MEMORY. THE DIAGNOSTIC SUPERVISOR CODING OCCUPIES 6.25K OF THE UPPER PART OF MEMORY JUST BELOW THE CNDP+ MONITOR WHICH RESIDES IN THE UPPERMOST 1.5K OF MEMORY SPACE.

1.1.2 DIAGNOSTIC INFORMATION

THE RL11/RLV11 CONTROLLER TEST (PART 2) IS A KXT-11 BASED PROGRAM THAT WILL TEST THE CONTROLLER. IT COMPLEMENTS PART 1 BY EXTENDING THE TEST COVERAGE TO INCLUDE WRITE DATA, READ DATA, WRITE CHECK AND READ DATA WITHOUT HEADER COMPARE. IT IS AIMED AT FULLY TESTING THE CONTROLLER IN THESE AREAS, BUT BY DEFAULT ALSO EXERCISES THE DRIVE.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

- SBC-11/21+ PROCESSOR, 28KW MEMORY, JUMPERED FOR MEMORY MAP 0
- 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'

- CNDP+ (XXDP+) LOAD DEVICE (RL02, RX02, ETC.)
- LINE PRINTER (OPTIONAL)

1.2.2 SOFTWARE REQUIREMENTS

CNRLHA RL11/RLV11 CTLR TEST 2

1.3 RELATED DOCUMENTS AND STANDARDS

RL01 DISK SUBSYSTEM USER'S GUIDE (EK-RL01-UG-002)
XXDP+/SUPERVISOR USER'S MANUAL

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

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

CVRLAB0	RLV11 RL01 DISKLESS TEST (RLV11 ONLY)
CNRLGA0	RL11/RLV11 RL01/02 CONTROLLER TEST (PART 1)

1.5 ASSUMPTIONS

THE HARDWARE OTHER THAN THE RL01/02 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 FIVE STEPS OF EXECUTION

THIS DIAGNOSTIC PROGRAM SHOULD BE LOADED AND STARTED USING NORMAL XXDP+ PROCEDURES. START THE EXECUTION OF THE CNDP+ MONITOR BY USING THE APPROPRIATE BOOTSTRAP PROGRAM. THE MONITOR WILL PRINT A MESSAGE IDENTIFYING ITSELF AND REQUESTING THAT THE CURRENT DATE BE ENTERED. AN EXAMPLE OF THIS MESSAGE IS GIVEN BELOW FOR THE CNDP+ MONITOR:

```
CNMDYAO CNDP+ DY MONITOR
BOOTED VIA UNIT 0
ENTER DATE (DD-MMM-YY):
```

AFTER THE DATE HAS BEEN ACCEPTED BY THE MONITOR THE RESTART ADDRESS OF THE MONITOR IS PRINTED. THEN THE FOLLOWING TWO QUESTIONS ARE ASKED:

```
50 HZ ? N
LSI ?  N
```

THE DEFAULTS ARE BOTH "NO". TYPE "R" AND THE PROGRAM NAME TO RUN THE PROGRAM. DO NOT TYPE THE EXTENSION.

WHEN THIS DIAGNOSTIC IS STARTED THE FOLLOWING 5 STEPS WILL OCCUR:

```
**********
* STEP 1 *
**********
```

THE DIAGNOSTIC WILL ISSUE THE PROMPT "DR>". FROM THIS POINT UNTIL THE TIME WHEN YOU RESTART CNDP+, YOU WILL BE TALKING TO THE DIAGNOSTIC, NOT CNDP+. WE WILL REFER TO THE PRESENCE OF THIS PROMPT AS BEING IN DIAGNOSTIC COMMAND MODE, AS OPPOSED TO CNDP+ COMMAND MODE.

AT THIS POINT YOU WILL ENTER A "START" COMMAND. THIS IS NOT THE SAME AS THE CNDP+ "START" COMMAND, WHICH YOU ALREADY ISSUED IN RESPONSE TO THE CNDP+ 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 "DR>" 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:

PNT	PRINT NUMBER OF TEST BEING EXECUTED
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 2 *
****

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 3 *
****

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 4 *
****

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 5 *
****

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 DR>).

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 OCCURRED.

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 1, 2, 3, 4, AND 5 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 OCCURRED. 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 ON ERROR).

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.

```
433
434                     WHEN YOU'VE SEEN ENOUGH, YOU MAY HIT CONTROL/C.  THIS WILL TAKE YOU
435                     OUT  OF  THE LOOP AND PUT YOU BACK INTO COMMAND MODE.  YOU NOW HAVE
436                     THREE CHOICES:
437
438                         1.  START
439
440                         2.  RESTART
441
442                         3.  CONTINUE
443
444
445                     LET'S SAY YOU'VE REPAIRED THE DEFECT FOUND ABOVE AND WANT TO FINISH
446                     RUNNING THE DIAGNOSTIC.  YOU WOULD TYPE
447
448                         CON/FLAGS:HOE:IER=0:LOE=0
449
450
451                     THIS WILL RESTORE THE FLAGS TO THEIR  ORIGINAL  VALUES  AND  RESUME
452                     EXECUTION  AT  THE  BEGINNING OF THE HARDWARE TEST YOU WERE IN.  IF
453                     THE ERROR DOES NOT RECUR, THE EXECUTION WILL FLOW RIGHT ON THRU  TO
454                     THE NEXT ERROR OR TO END OF PASS.
455
456                     IF AT END OF PASS YOU WANT TO RUN THE DIAGNOSTIC  AGAIN,  YOU  HAVE
457                     TWO CHOICES:
458
459                         1.  START
460
461                         2.  RESTART
462
463
464                     YOU WOULD CHOOSE ONE, DEPENDING ON WHETHER YOU WANTED TO ANSWER THE
465                     HARDWARE QUESTIONS AGAIN.
466
```

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

```
                                                    BY
                                                    WHOM
                                                    ENTERED:
                                                    -------
.R NRLHAO                                           O
DRS LOADED                                          D
DIAG. RUN-TIME SERVICES REV. D APR-79               D
CNRLH-A-O                                           D
CNRLH TESTS WRITE DATA, READ DATA,                  D
AND WRITE CHECK OPERATIONS
UNIT IS RL01, RL02                                  D
DR>STA/PASS:1/FLAGS:HOE                             D,O

CHANGE HW (L) ? Y                                   D,O

# UNITS (D) ? 2                                     D,O

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

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

DROP ON ERROR LIMIT (L) N ?
COMPARE DATA ON DCK (L) N ?


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

DR>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
**********************************************************
```

```
522
523                         ^C                                  0
524
525                         DR>CON/FLAGS:HOE:IER:LOE=0          D,0
526
527                         CHANGE SW (L) ? N                   D,0
528
529                         CNRLH EOP 1                         D
530                         ^C
531
532                         DR>RESTART/PASS:1                   D,0
533
534                         CHANGE SW (L) ? N                   D,0
535                         ----------------
536                         ----------------
537                         ----------------
538                         ----------------
539
540
541               2.2       CHAIN MODE OPERATION
542                         ----- ---- ---------
543
544                         CHAIN MODE OPERATION CONSISTS OF THE SEQUENTIAL EXECUTION OF
545                         PROGRAMS WITHOUT OPERATOR INTERVENTION.  ONLY PROGRAMS THAT HAVE
546                         BEEN MODIFIED TO RUN IN CHAIN MODE CAN BE CHAINED.  CHAINABLE
547                         PROGRAMS ARE IDENTIFIED IN THE DIRECTORY BY A BIC EXTENSION.
548
549                         TO RUN CHAIN MODE, THE CNDP+ MONITOR USES AN ASCII FILE (KNOWN AS A
550                         CHAIN FILE) LISTING THE PROGRAMS TO BE RUN AND THE NUMBER OF PASSES
551                         EACH PROGRAM SHOULD RUN.  THIS FILE MUST BE ON THE SYSTEM DEVICE.
552
553                         A CHAIN FILE MAY BE GENERATED BY USE OF THE XTECO TEXT EDITOR.
554                         THIS FILE MUST HAVE A CCC EXTENSION.  THE CHAIN FILE MAY CONTAIN
555                         ANY OF THE COMMANDS SUPPORTED BY THE CNDP+ MONITOR.  THE COMMANDS
556                         IN THE ASCII FILE ARE EXECUTED IN THE ORDER IN WHICH THEY ARE
557                         ENCOUNTERED.
558
559                         TO EXECUTE A CHAIN FILE THE USER TYPES:
560
561                               C FILNAM <CR> OR
562                               C FILNAM/QV  <CR>
563
564                         IN THE FIRST CASE THE PASS COUNT SPECIFIED IN THE CHAIN FILE IS
565                         USED BY THE CNDP+ MONITOR TO DETERMINE THE NUMBER OF PASSES TO
566                         EXECUTE EACH PROGRAM.  IN THE SECOND CASE THE PASS COUNT IS NOT
567                         USED AND EACH PROGRAM IS EXECUTED ONLY ONCE.  THE /QV SWITCH
568                         PROVIDES A SINGLE EXECUTION MODE OF OPERATION OF QUICK VERIFY.
569
570                         WHEN PROGRAMS ARE RUN IN CHAIN MODE, THE SOFTWARE SWITCH REGISTER
571                         SHOULD BE SET TO 000000.  THE CNDP+ MONITOR PRINTS EACH COMMAND
572                         TAKEN FROM THE CHAIN FILE AND THEN EXECUTES THE COMMAND.  WHEN THE
573                         LAST COMMAND OTHER THAN ANOTHER C COMMAND HAS BEEN EXECUTED THE
574                         CNDP+ MONITOR TERMINATES CHAIN MODE AND TYPES A PROMPT (.).  IT IS
575                         READY TO ACCEPT ANOTHER COMMAND FROM THE CONSOLE.  IF THE LAST
576                         COMMAND IS ANOTHER C COMMAND, THE CHAIN MODE WILL CONTINUE AND THE
577                         CHAIN FILE SPECIFIED BY THIS NEW C COMMAND WILL BE USED.
```

IF THE USER WISHES TO TERMINATE CHAIN MODE BEFORE ITS NORMAL TERMINATION HE MAY DO SO BY TYPING A CONTROL/C. HOWEVER, THE MONITOR WILL NOT ABORT THE CHAIN MODE UNTIL IT RECEIVES PROGRAM CONTROL FROM THE PROGRAM CURRENTLY RUNNING.

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 EXIT
2. DIAGNOSTIC HAS FINISHED ALL ITS REQUESTED PASSES	START RESTART PRINT DISPLAY FLAGS ZFLAGS EXIT
3. OPERATOR INTERRUPTED THE DIAGNOSTIC WITH CTRL/C	START RESTART CONTINUE PRINT DISPLAY FLAGS ZFLAGS EXIT
4. AN ERROR WAS ENCOUNTERED WITH THE HOE FLAG SET SET	START RESTART CONTINUE PROCEED PRINT DISPLAY FLAGS ZFLAGS EXIT

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

```
687
688                         BOE BELL ON ERROR
689
690                         UAM RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS
691
692                         ISR INHIBIT STATISTICAL REPORTS
693
694                         IDU INHIBIT DROPPING OF UNITS BY DIAGNOSTIC
695
696                         ADR EXECUTE AUTODROP CODE
697
698                         LOT LOOP ON TEST
699
700                         EVL EVALUATE
701
702
703                     THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE  EQUATED  TO  0  ARE
704                     CLEARED.   A FLAG NOT SPECIFIED IS CLEARED.  IF THE FLAGS SWITCH IS
705                     NOT GIVEN ALL FLAGS ARE CLEARED.
706
707                     "EOP-INCR" IS A DECIMAL NUMBER INDICATING HOW OFTEN  (IN  TERMS  OF
708                     PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE PRINTED.  THE
709                     DEFAULT IS AT THE END OF EVERY PASS.
710
711                     ********************************************************************************
712                     RES(TART)/TEST:TEST-LIST/PASS:PASS-CNT/FLAGS:FLAG-LIST/EOP:EOP-INCR/UNITS:UNIT-LIST
713                     ********************************************************************************
714
715                     THE DIAGNOSTIC IN CORE IS EXECUTED IN ACCORDANCE WITH THE  SWITCHES
716                     SPECIFIED.   HOWEVER,  NEW  "P-TABLES" ARE NOT BUILT.  INSTEAD, THE
717                     ONES IN CORE ARE USED.
718
719                     THE QUESTION "CHANGE SW?" IS ASKED AND THE ANSWERS GIVEN BECOME THE
720                     NEW  DEFAULTS.  THE COMMAND MAY BE ISSUED WHEN COMAND MODE HAS BEEN
721                     ENTERED  VIA  A)  DIAGNOSTIC  IS  FINISHED  B)  HALT  ON  ERROR  C)
722                     CONTROL/C.
723
724                     THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:
725
726                         1.  "UNIT-LIST" IS A SEQUENCE OF LOGICAL UNIT  NUMBERS  RANGING
727                             FROM 1 THRU N (N = NUMBER OF UNITS BEING TESTED) SPECIFYING
728                             WHICH UNITS ARE TO BE  TESTED.   THE  LOGICAL  UNIT  NUMBER
729                             DESIGNATES  THE  POSITION OF THE P-TABLE IN CORE, ACCORDING
730                             TO THE ORDER IN WHICH THEY WERE BUILT.  THE UNITS SPECIFIED
731                             MUST  NOT  HAVE  BEEN DROPPED BY THE OPERATOR DROP COMMAND.
732                             THE UNIT-LIST DEFAULTS TO "ALL THAT HAVE NOT  BEEN  DROPPED
733                             BY  OPERATOR  COMMAND".   THE EFFECT OF THE UNIT-LIST LASTS
734                             UNTIL THE NEXT START (WHERE IT IS  AUTOMATICALLY  RESET  TO
735                             "ALL") OR THE NEXT RESTART.
736
737                         2.  ALL UNSPECIFIED FLAG SETTINGS ARE UNCHANGED.
738
```

```
                        ;*************************************************
                        ;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.  DEFAULT FOR PASS-CNT IS THE UNSATISFIED PASS-CNT FROM THE
                        ;        PREVIOUS START OR RESTART

                        ;    2.  UNSPECIFIED FLAG SETTINGS ARE UNCHANGED


                        ;*****************************
                        ;PRO(CEED)/FLAGS:<FLAC ' IST>
                        ;*****************************

                        ;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


                        ;****
                        ;EXIT
                        ;****

                        ;RETURN TO CNDP+ PROMPT MODE.


                        ;************************
                        ;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.
```

```
792
793                             **********
794                             ADD/UNITS:UNIT-LIST
795                             **********
796
797                             THE UNITS SPECIFIED ARE ADDED BACK (THEY MUST HAVE BEEN  PREVIOUSLY
798                             DROPPED  BY  THE DROP COMMAND) TO THE TEST SEQUENCE.  AN ADD CANNOT
799                             BE FOLLOWED BY A PROCEED.
800
801
802                             *******
803                             PRI(NT)
804                             *******
805
806                             ALL STATISTICS TABLES ACCUMULATED BY THE  DIAGNOSTIC  ARE  PRINTED.
807                             THE ISR (INHIBIT STATISTICAL REPORTING) FLAG IS CLEARED.
808
809
810                             ****************************
811                             DIS(PLAY)/UNITS:<UNIT-LIST>
812                             ****************************
813
814                             THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED  OUT  IN
815                             THE FORMAT IN WHICH THEY WERE ENTERED.  ANY UNITS THAT WERE DROPPED
816                             BY THE OPERATOR "DROP" COMMAND ARE SO DESIGNATED.
817
818
819                             *******
820                             FLA(GS)
821                             *******
822
823                             THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.
824
825
826                             ********
827                             ZFL(AGS)
828                             ********
829
830                             ALL FLAGS ARE CLEARED.
831
832
833                     2.4         EXTENDED P-TABLE DIALOGUE
834                                 -------- ------- --------
835
836                             THE FULL CAPABILITY OF THE HARDWARE DIALOGUE  IS  REVEALED  BY  THE
837                             FOLLOWING DISCUSSION OF WHAT HAPPENS INTERNALLY.
838
839                             AS SOON AS THE QUESTION "# UNITS?" IS ANSWERED (WITH THE NUMBER N),
840                             SPACE  IN  CORE IS ALLOCATED FOR "N" P-TABLES.  ALL OF THE P-TABLES
841                             ARE OF THE SAME FORMAT, AND THERE IS  A  ONE-TO-ONE  CORRESPONDENCE
842                             BETWEEN  THE  HARDWARE  PARAMETER  QUESTIONS  AND  THE SLOTS IN THE
843                             P-TABLE FORMAT.
```

```
845
846                     IN GIVING A STRING OF VALUES, COMMAS WITHOUT INTERVENING VALUES MAY
847                     BE USED TO INDICATE A REPETITION OF THE LAST NAMED VALUE.
848
849                     A STRING OF VALUES MAY BE GIVEN AS A RANGE (6-10 FOR EXAMPLE).  IF
850                     THE  VALUES  REPRESENT  PURE  NUMERICAL  DATA,  THIS  SAMPLE  RANGE
851                     TRANSLATES TO THE STRING 6,7,8,9,10 (AN INCREMENT OF  1).  IF  THE
852                     VALUES  ARE  ADDRESSES,  THE  SAMPLE RANGE TRANSLATES TO THE STRING
853                     6,8,10 (AN INCREMENT OF 2).
854
855                     NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO  CONSTRUCT  A
856                     SET  OF  P-TABLES.  ASSUME THAT WE HAVE 8 RL UNITS, AND THAT THERE
857                     ARE FIVE (5) HARDWARE PARAMETERS FOR EACH (5 SLOTS IN THE  P-TABLE,
858                     5 HARDWARE QUESTIONS IN THE DIALOGUE).
859
860                     FOLLOWING IS THE DIALOGUE FOR  THIS  8  RLOX  DRIVE  SYSTEM.  THIS
861                     SYSTEM  HAS  TWO  (2)  RL11  TYPE  CONTROLLERS ALL TO BE SET AT "BR
862                     LEVEL" 5.  THE FIRST 4 DRIVES ARE RL01'S AND THE LAST 4 DRIVES  ARE
863                     RL02'S (ON THE SECOND CONTROLLER):
864
865                     # UNITS (D) ? 8
866
867                     UNIT 0
868                     RL11 (L) Y ?
869                     BUS ADDDRESS (O) 174400 ?
870                     VECTOR (O) 160 ?
871                     DRIVE TYPE = RL01 (L) Y ?
872                     BR LEVEL (O) 5 ?
873                     DRIVE (O) 0 ? 0-3
874
875                     UNIT 4
876                     RL11 (L) Y ?
877                     BUS ADDRESS (O) 174400 ? 175400
878                     VECTOR (O) 160 ? 164
879                     DRIVE TYPE = RL01 (L) Y ? N
880                     BR LEVEL (O) 5 ?
881                     DRIVE (O) 0 ? 0-3
882
883                     THE FIRST TIME THRU THE P-TABLE QUESTIONS THE  DEFAULT  VALUES  ARE
884                     USED  FOR  THE  CONTROLLER  TYPE  (QUESTION #1), CSR ADDRESS OF THE
885                     CONTROLLER  (QUESTION  #2),  THE  CONTROLLER  VECTOR  ASSIGNMENT
886                     (QUESTION  #3),  THE  DRIVE  TYPE (QUESTION #4), AND THE "BR LEVEL"
887                     (QUESTION #5).  THE ACTUAL UNIT NUMBERS OF THE RL01'S FOR  QUESTION
888                     #6 WERE ASSIGNED 0 THRU 3 FOR THE FIRST 4 P-TABLE SLOTS.
889
890                     THE SECOND TIME THRU THE P-TABLE QUESTIONS (FOR THE RL02 ASSIGNMENT
891                     ON  THE  SECOND CONTROLLER), THE FIRST QUESTION DEFAULTED TO "RL11"
892                     TYPE CONTROLLER.  THE SECOND QUESTION WAS ANSWERED TO  REFLECT  THE
893                     CHANGE IN CSR ADDRESS FOR THE RL02 CONTROLLER (175400).  THE SECOND
894                     CONTROLLER'S VECTOR WAS ALSO CHANGED TO 164 IN  QUESTION  #3.  THE
895                     RL02  TEST  UNIT NUMBERS WERE ASSIGNED VALUES 0 TO 3 IN QUESTION #4
896                     AND THE DRIVE TYPE WAS SET FOR RL02'S FOR THE REMAINING 4 UNITS  IN
897                     QUESTION  #4.  QUESTION #5 WAS DEFAULTED USING THE "BR LEVEL" FROM
898                     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 RESPONSE.

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 TYPE = RL01 (L) ?

ANSWER NO (N) IF DRIVE IS AN RL02

BR LEVEL (O) 5?

ANSWER WITH THE INTERRUPT PRIORITY OF THE CONTROLLER.

DRIVE (O) 0?

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

2.6 SOFTWARE PARAMETERS

THE FOLLOWING QUESTIONS ARE ASKED IF REQUESTED ON A START, RESTART, OR CONTINUE. THEY ALLOW FLEXIBILITY 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.

"DROP ON ERROR LIMIT (L) Y?"

```
953
954                     TO ALLOW THE UNIT TO BE DROPPED  ONCE  A  PREDETERMINED  NUMBER  OF
955                     ERRORS ARE ENCOUNTERED.
956
957                             ANSWER Y OR N
958
959                             "ERROR LIMIT (D) 10?"
960
961                     NUMBER OF ERRORS ALLOWED BEFORE DROPPING UNIT.
962
963                             ANSWER 1 TO 65K
964
965                             "COMPARE DATA ON DCK (L) N?"
966
967                     WHEN A DATA CHECK IS ENCOUNTERED AND DATA IS KNOWN, ALLOW AN INCORE
968                     COMPARISON OF DATA.
969
970                             ANSWER Y OR N
971
972                             "# OF WORDS IN ERROR REPORTED (D) 3?  "
973
974                     NUMBER OF MISCOMPARES TO BE PRINTED ON CONSOLE DEVICE.
975
976                             ANSWER 0 - 128
977
978
979                 3.0         ERROR INFORMATION
980                             ----- -----------
981
982                     ALL ERROR INFORMATION IS PRINTED  ON  THE  CONSOLE  DEVICE.  ERROR
983                     REPORTS  ARE  AIMED  AT BEING SELF EXPLANATORY.  THE GENERAL FORMAT
984                     IS:
985
986                       CNRLH  XXX    ERR  YYYYY   TST  ZZZ   SUB  PPP   PC:  RRRRRR
987
988                       WHERE:
989                             XXX     IS SFT - SOFT ERROR
990                                        HRD - HARD ERROR
991                                     DV FAT - DEVICE FATAL ERROR
992                                    SYS FAT - SYSTEM FATAL ERROR
993                             YYYYY    IS THE ERROR NUMBER
994                             ZZZ      IS THE TEST NUMBER
995                             PPP      IS THE SUBTEST NUMBER
996                             RRRRRR   IS THE PROGRAM LISTING LOCATION
997
998                       ERRORS GIVE THE REGISTER CONTENTS BEFORE AND AFTER THE ERROR
999                       ALONG WITH A ONE LINE DESCRIPTION AND RELEVENT DATA.
1000
1001                      EXAMPLE:
1002
1003                      ONE LINE DESCRIPTION
1004                      (OPTIONAL SECOND LINE)
1005                      (OPTIONAL THIRD LINE)
1006                      BEFORE COMMAND: CS:XXXXXX BA:XXXXXX DA:XXXXXX MP:XXXXXX
1007                      TIME OF ERROR:  CS:XXXXXX BA:XXXXXX DA:XXXXXX MP:XXXXXX XXXXXX XXXXXX
```

```
1009
1010                                3.1         ERROR HALTS
1011                                            ----- -----
1012
1013                                ERROR HALTS ARE SUPPORTED PER DESCRIBED  IN  THE  PREVIOUS  SECTION
1014                                WITH /FLAG:HOE.  THERE ARE NO OTHER HALTS.
1015
1016
1017                                4.0         PERFORMANCE AND PROGRESS REPORTS
1018                                            ----------- --- -------- -------
1019
1020
1021                                4.1         PERFORMANCE REPORTS
1022                                            ----------- -------
1023
1024                                THIS PROGRAM WILL NOT GIVE ANY PERFORMANCE REPORTS.
1025
1026
1027                                4.2         PROGRESS REPORTS
1028                                            -------- -------
1029
1030                                THIS PROGRAM WILL NOT GIVE ANY PROGRESS REPORTS.
1031
1032
1033                                5.0         DEVICE INFORMATION TABLES
1034                                            ------ ----------- ------
1035
1036                                THE RL11/RLV11 CONTROLLER HAS THE FOLLOWING FOUR(4)  REGISTERS  FOR
1037                                CONTROL OF THE SUBSYSTEM.
1038
1039                                            RLCS - CONTROL AND STATUS REGISTER (XXXXX0)
1040                                            ------------------------------------------
1041
1042                                BIT 15 - COMPOSITE ERROR
1043                                BIT 14 - DRIVE ERROR
1044                                BIT 13 - NON EXISTANT MEMORY ERROR
1045                                BIT 12 - HEADER NOT FOUND (WITH BIT 10 SET)
1046                                       - DATA LATE (WITH BIT 10 CLEAR)
1047                                BIT 11 - HEADER CRC (WITH BIT 10 SET)
1048                                       - DATA CRC (WITH BIT 10 CLEAR)
1049                                BIT 10 - OPERATION INCOMPLETE
1050                                BIT  9/8 - DRIVE SELECT (0-3)
1051                                BIT  7 - CONTROLLER READY
1052                                BIT  6 - INTERRUPT ENABLE
1053                                BIT  5 - EXTENDED BUS ADDRESS (BIT 17)
1054                                BIT  4 - EXTENDED BUS ADDRESS (BIT 16)
1055                                BIT 3-1 - FUNCTION CODE
1056                                        0 - NOP (PDP-11) MAINT (LSI-11)
1057                                        1 - WRITE CHECK
1058                                        2 - GET DRIVE STATUS
1059                                        3 - SEEK
1060                                        4 - READ HEADER
1061                                        5 - WRITE DATA
1062                                        6 - READ DATA
1063                                        7 - READ WITHOUT HEADER COMPARE
```

```
1065
1066                                    BIT  0 - DRIVE READY
1067
1068                                    RLBA - BUS ADDRESS REGISTER (XXXXX2)
1069                                    -----------------------------------
1070
1071                                    BITS 15-1 BUS ADDRESS OF DATA TRANSFER
1072                                    BIT 0 SHOULD BE 0
1073
1074                                    RLDA - DISK ADDRESS REGISTER (XXXXX4)
1075                                    -----------------------------------
1076
1077                                    FOR READ/WRITE FUNCTIONS
1078                                    ------------------------
1079
1080                                    BIT 15-7 - CYLINDER ADDRESS FOR TRANSFER
1081                                    BIT 6 - SURFACE FOR TRANSFER
1082                                    BIT 5-0 - SECTOR FOR TRANSFER (1-40.)
1083
1084                                    FOR SEEK FUNCTION
1085                                    -----------------
1086
1087                                    BIT 15-7 - DIFFERENCE TO NEW CYLINDER
1088                                    BIT 6-5 - MUST BE ZERO (0)
1089                                    BIT 4 - SURFACE (0=UPPER, 1=LOWER)
1090                                    BIT 3 - MUST BE ZERO (0)
1091                                    BIT 2 - SEEK DIRECTION( 1=IN / 0=OUT )
1092                                    BIT 1 - MUST BE ZERO (0)
1093                                    BIT 0 - MUST BE ONE (1)
1094
1095                                    FOR GET STATUS FUNCTION
1096                                    -----------------------
1097
1098                                    BIT 15-4 - IGNORED SHOULD BE ZERO (0)
1099                                    BIT 3 - DRIVE RESET
1100                                    BIT 2 - MUST BE ZERO (0)
1101                                    BIT 1 - MUST BE ONE (1)
1102                                    BIT 0 - MUST BE ONE (1)
1103
1104                                    RLMP - MULTIPURPOSE REGISTER
1105                                    ----------------------------
1106
1107                                    FOR READ/WRITE FUNCTION
1108                                    -----------------------
1109
1110                                    BIT 15 - 0 - WORD COUNT (TWO'S COMPLIMENT)
1111
1112                                    FOR READ HEADER FUNCTION
1113                                    ------------------------
1114
1115                                    BIT 15-0 - DISK HEADER OF SECTOR (FIRST READ)
1116                                             - ZERO WORD (SECOND READ)
1117                                             - HEADER CRC (THIRD READ)
```

```
1119
1120                                    FOR GET STATUS FUNCTION
1121                                    -----------------------
1122
1123                                    HAS DRIVE STATUS
1124
1125                                    BIT 15 - WRITE DATA ERROR
1126                                    BIT 14 - CURRENT HEAD ERROR (CHE)
1127                                    BIT 13 - WRITE LOCK STATUS (WL)
1128                                    BIT 12 - SEEK TIME OUT (SKTO)
1129                                    BIT 11 - SPIN ERROR (SPE)
1130                                    BIT 10 - WRITE GATE ERROR (WGE)
1131                                    BIT  9 - VOLUME CHECK (VC)
1132                                    BIT  8 - DRIVE SELECT ERROR (DSE)
1133                                    BIT  7 - DRIVE TYPE IS RL02 IF SET
1134                                    BIT  6 - SURFACE (0=UPPPER, 1=LOWER)
1135                                    BIT  5 - COVER OPEN
1136                                    BIT  4 - HEADS HOME
1137                                    BIT  3 - BRUSHES HOME
1138                                    BIT 2-0 -STATE BITS
1139                                          0 - LOAD STATE
1140                                          1 - SPIN UP
1141                                          2 - BRUSH CYCLE
1142                                          3 - LOAD HEADS
1143                                          4 - SEEK - TRACK COUNTING
1144                                          5 - SEEK - LINEAR MODE
1145                                          6 - UNLOAD HEADS
1146                                          7 - SPIN DOWN
1147
1148
1149                            6.0           TEST SUMMARIES
1150                                          ---- ---------
1151
1152                                    TEST 1 - WRITE FUNCTION
1153                                    ******
1154
1155                                          THIS TEST WILL VERIFY THAT THE WRITE FUNCTION WILL RESET
1156                                          CONTROLLER READY AND POST NO ERRORS.
1157
1158
1159                                    TEST 2 - WRITE FUNCTION INTERRUPT
1160                                    ******
1161
1162                                          THIS TEST WILL VERIFY THAT THE WRITE FUNCTION WILL GENERATE
1163                                          AN INTERRUPT ON COMPLETION.
1164
1165
1166                                    TEST 3 - PROPER INCREMENT OF RLBA ON WRITE
1167                                    ******
1168
1169                                          THIS TEST WILL VERIFY THAT THE BUS ADDRESS REGISTER INCREMENTS
1170                                          PROPERLY ON A WRITE FUNCTION.
```

```
1172
1173                    TEST 4 - PROPER INCREMENT OF RLDA ON WRITE
1174                    ******
1175
1176                            THIS TEST WILL VERIFY THAT THE DISK ADDRESS REGISTER INCREMENTS
1177                            PROPERLY ON A WRITE FUNCTION.
1178
1179
1180                    TEST 5 - FORCE HEADER NOT FOUND WITH WRITE
1181                    ******
1182
1183                            THIS TEST WILL FORCE A HEADER NOT FOUND ERROR ON A WRITE.  THE
1184                            RLDA IS SET UP TO LOOK FOR SECTOR 40, A WRITE IS THEN ISSUED.
1185                            THE HEADER NOT FOUND ERROR SHOULD THEN SET.
1186
1187
1188                    TEST 6 - FORCE HEADER NOT FOUND WITH WRITE INTERRUPT
1189                    ******
1190
1191                            THIS TEST WILL FORCE A HEADER NOT FOUND ERROR UNDER INTERRUPT
1192                            CONTROL.  HEADER NOT FOUND IS FORCED BY SETTIING SECTOR 40 OF
1193                            RLDA AND ISSUING A WRITE.
1194
1195
1196                    TEST 7 - CHECK OPI TIME WITH HNF
1197                    ******
1198
1199                            (THIS TEST HAS BEEN REMOVED FROM CNRLHA). -- JSD REV A
1200
1201                            THIS TEST WILL TIME THE SETTING OF HNF (OPI) FROM ISSUANCE.
1202                            THIS IS DONE BY ISSUING A WRITE TO SECTOR 40.  THE TIME OF OPI
1203                            SHOULD BE AROUND 200 MILLISECONDS.
1204
1205
1206                    TEST 8 - MULTIPLE SECTOR TRANSFER ON WRITE
1207                    ******
1208
1209                            THIS TEST THE ABILITY FOR THE WRITE FUNCTION TO WRITE MORE THAN
1210                            ONE SECTOR.  WE SET UP FOR A TWO SECTOR WRITE.
1211
1212
1213                    TEST 9 - CHECK DIRECTION OF WRITE NPR
1214                    ******
1215
1216                            THIS 1EST WILL VERIFY THAT THE NPR DIRECTION OF A WRITE FUNC-
1217                            TION IS FROM MEMORY TO THE CONTROLLER.  THIS IS DONE BY
1218                            WRITING A PATTERN IN MEMORY AND ISSUING A WRITE. THEN CHECKING
1219                            MEMORY TO VERIFY THAT IT DID NOT GET DISTURBED.
1220
1221
1222                    TEST 10 - CHECK FULL INCREMENT OF RLBA
1223                    *******
1224
1225                            THIS TEST WILL CHECK THAT THE RLBA CAN INCREMENT OF THE FULL
1226                            16 BIT RANGE.  THIS IS DONE BY ISSUING A ONE WORD WRITE TO
1227                            CHECK EACH BIT TOGGLE FROM 1-0 AND 0-1.  THIS IS DONE FROM 0
1228                            TO 177776 REGARDLESS OF MEMORY SIZE.
```

```
1230
1231                                    TEST 11 - BA BIT 16 INCREMENT
1232                                    *******
1233
1234                                            THIS TEST WILL CHECK THAT BUS ADDRESS BIT 16 WILL SET WHEN THE
1235                                            RLBA IS 177776.  AND THAT THE RLBA GOES TO 0.
1236
1237
1238                                    TEST 12 - BA BIT 17 INCREMENT
1239                                    *******
1240
1241                                            THIS TEST WILL CHECK THAT BUS ADDRESS BIT 17 WILL SET WHEN
1242                                            BIT 16 AND THE RLBA ARE SET.  THE RLBA AND BIT 16 ARE CHECKED
1243                                            TO GO TO ZERO.
1244
1245
1246                                    TEST 14 - READ NPR INTEGRITY
1247                                    *******
1248
1249                                            THIS TEST WILL VERIFY THAT THE READ FUNCTION WILL NOT CAUSE
1250                                            A BUS TRAP THEREFORE VERIFYING THE NPR LOGIC BETWEEN THE
1251                                            CONTROLLER AND PROCESSOR.
1252
1253
1254                                    TEST 13 - READ FUNCTION
1255                                    *******
1256
1257                                            THIS TEST WILL VERIFY THAT THE READ FUNCTION WILL RESET
1258                                            CONTROLLER READY AND POST NO ERRORS.
1259
1260
1261                                    TEST 14 - READ FUNCTION INTERRUPT
1262                                    *******
1263
1264                                            THIS TEST WILL VERIFY THAT THE READ FUNCTION WILL GENERATE
1265                                            AN INTERRUPT ON COMPLETION.
1266
1267
1268                                    TEST 15 - CHECK DIRECTION OF READ NPR
1269                                    *******
1270
1271                                            THIS TEST WILL VERIFY THAT THE NPR DIRECTION OF A READ FUNC-
1272                                            TION IS FROM CONTROLLER TO THE MEMORY.  THIS IS DONE BY
1273                                            WRITING A PATTERN IN MEMORY AND ISSUING A READ, THEN CHECKING
1274                                            MEMORY TO VERIFY THAT IT DID NOT GET DISTURBED.
1275
1276
1277                                    TEST 16 - PROPER INCREMENT OF RLBA ON READ
1278                                    *******
1279
1280                                            THIS TEST WILL VERIFY THAT THE BUS ADDRESS REGISTER INCREMENTS
1281                                            PROPERLY ON A READ FUNCTION.
```

```
1283
1284                    TEST 17 - PROPER INCREMENT OF RLDA ON READ
1285                    *******
1286
1287                            THIS TEST WILL VERIFY THAT THE DISK ADDRESS REGISTER INCREMENTS
1288                            PROPERLY ON A READ FUNCTION.
1289
1290
1291                    TEST 18 - FORCE HEADER NOT FOUND WITH READ
1292                    *******
1293
1294                            THIS TEST WILL FORCE A HEADER NOT FOUND ERROR ON A READ.  THE
1295                            RLDA IS SET UP TO LOOK FOR SECTOR 40, A READ IS THEN ISSUED.
1296                            THE HEADER NOT FOUND ERROR SHOULD THEN SET.
1297
1298
1299                    TEST 19 - FORCE HEADER NOT FOUND WITH READ INTERRUPT
1300                    *******
1301
1302                            THIS TEST WILL FORCE A HEADER NOT FOUND ERROR UNDER INTERRUPT
1303                            CONTROL.  HEADER NOT FOUND IS FORCED BY SETTING
1304                            SECTOR 40 OF RLDA AND ISSUING A READ.
1305
1306
1307                    TEST 20 - CHECK HEADER COMPARE LOGIC
1308                    *******
1309
1310                            THIS TEST WILL EXTENSIVELY CHECK THE CYLINDER AND HEAD BITS OF
1311                            THE HEADER WORD TO COMPARE CORRECTLY.  THIS IS DONE BY WALKING
1312                            AND GROWING 0'S AND 1'S THRU THE PROPER RLDA BITS AND ISSUING
1313                            READ TO SEE IF ALL BIT POSITIONS CAN COMPARE.
1314
1315
1316                    TEST 21 - MULTIPLE SECTOR TRANSFER ON READ
1317                    *******
1318
1319                            THIS TEST THE ABILITY FOR THE READ FUNCTION TO WRITE MORE THAN
1320                            ONE SECTOR.  WE SET UP FOR A TWO SECTOR READ.
1321
1322
1323                    TEST 22 - FORCE HNF AT END OF TRACK
1324                    *******
1325
1326                            THIS TEST WILL CHECK THE ABILITY TO DETECT HEADER NOT FOUND
1327                            AT THE END OF A TRACK.  THIS DONE BY SETTING UP FOR A TWO
1328                            SECTOR READ AT SECTOR 39.
1329
1330
1331                    TEST 23 - FORCE NON-EXISTENT MEMORY ERROR
1332                    *******
1333
1334                            THIS TEST WILL CHECK THAT THE NON-EXISTANT MEMORY ERROR (NXM)
1335                            CAN SET.  WE WILL ISSUE A READ TO THE MAXIMUM ADDRESS AND
1336                            EXPECT A NXM ERROR.  (THIS TEST WILL NOT BE DONE ON A 128K
1337                            MACHINE.)
```

```
1339
1340                    TEST 24 - FORCE NXM UNDER INTERRUPT
1341                    ********
1342
1343                            THIS TEST WILL ATTEMPT TO FORCE AN INTERRUPT VIA NXM.  (THIS
1344                            TEST WILL NOT BE DONE ONA 128K MACHINE.)
1345
1346
1347                    TEST 25 - CHECK READ WRITE LOOP
1348                    ********
1349
1350                            THIS TEST WILL WRITE A PATTERN TO SECTOR 0 AND TRY TO RECOVER
1351                            IT WITH A WRITE.
1352
1353
1354                    TEST 26 - CHECK OF SILO LINES
1355                    ********
1356
1357                            THIS TEST WILL CHECK THAT WE CAN WRITE AND READ UNIQUE BIT
1358                            PATTERNS VERIFY THAT THE LINES ON THE SILO ARE NOT STUCK OR
1359                            TIED TOGETHER.  THIS IS DONE WITH WALKING AND GROWING 0'S AND
1360                            1'S.
1361
1362
1363                    TEST 27 - CHECK THROUGHPUT OF SILO
1364                    ********
1365
1366                            THIS TEST WILL ATTEMPT TO CHECK THAT THE FALL THROUGH OF THE
1367                            SILO IS WORKING CORRECTLY.  WE WRITE A SECTOR OF 128 UNIQUE
1368                            PATTERNS AND READ IT BACK CHECKING THAT EACH LOCATION IS UNIQUE
1369                            AND CORRECT.
1370
1371
1372                    TEST 28 - CHECK ZERO FILL ON WRITE
1373                    ********
1374
1375                            THIS TEST WILL CHECK THE ABILITY OF THE CONTROLLER TO FILL THE
1376                            REMAINING SECTOR WITH ZEROS ON A WRITE.  WE WRITE A SECTOR WITH
1377                            FROM 1 TO 127 WORDS,  READ IT BACK AND VERIFY THAT THE NON
1378                            WRITTEN WORDS ARE ZERO.
1379
1380
1381                    TEST 29 - CHECK SECTOR BITS ON HEADER COMPARE
1382                    ********
1383
1384                            THIS TEST WILL CHECK THAT THE SECTOR BITS CAN COMPARE CORRECTLY.
1385                            THIS IS DONE BY WRITING THE SECTORS ADDRESS INTO THE SECTOR
1386                            FOR A FULL TRACK.  EACH SECTOR IS READ TO VERIFY THE SECTOR HAS
1387                            THE CORRECT DATA, IF NOT THEN THE SECTOR BITS ARE NOT COMPARING
1388                            CORRECTLY.
1389
1390                    TEST 30 - WRITE CHECK NPR INTEGRITY
1391                    ********
1392
1393                            THIS TEST WILL CHECK THAT THE WRITE CHECK WILL FUNCTION WITHOUT
1394                            CAUSING A BUS TRAP.  TEST IS SET UP TO HANDLE BUS TRAPS.
```

```
1396
1397                                    TEST 31 - WRITE CHECK FUNCTION
1398                                    *******
1399
1400                                            THIS TEST WILL CHECK THAT A WRITE CHECK FUNCTION WILL COMPLETE
1401                                            WITH THE SPECIFIED TIME WITHOUT POSTING ERRORS.
1402
1403
1404                                    TEST 32 - WRITE CHECK FUNCTION INTERRUPT
1405                                    *******
1406
1407                                            THIS TEST WILL CHECK THAT AN INTERRUPT CAN BE GENERATED FROM
1408                                            ISSUING A WRITE CHECK.
1409
1410
1411                                    TEST 33 - PROPER INCREMENT OF RLBA ON WRITE CHECK
1412                                    *******
1413
1414                                            THIS TEST WILL CHECK THAT THE RLBA INCREMENTS PROPERLY DURING A
1415                                            WRITE CHECK.
1416
1417
1418                                    TEST 34 - PROPER INCREMENT OF RLDA ON WRITE CHECK
1419                                    *******
1420
1421                                            THIS TEST WILL CHECK THAT THE RLDA INCREMENTS PROPERLY DURING A
1422                                            WRITE CHECK.
1423
1424
1425                                    TEST 35 - MULTIPLE SECTOR WRITE CHECK
1426                                    *******
1427
1428                                            THIS TEST WILL CHECK THAT WE CAN WRITE CHECK MORE THAN ONE SECTOR
1429                                            AT A TIME.
1430
1431
1432                                    TEST 36 - FORCE DCK WITH WRITE CHECK
1433                                    *******
1434
1435                                            THIS TEST WILL CHECK THAT WE CAN DETECT A DCK DURING A WRITE
1436                                            CHECK.  THIS IS DONE BY MODIFYING MEMORY BETWEEN A WRITE AND A
1437                                            WRITE CHECK.
1438
1439
1440                                    TEST 37 - FORCE DCK WITH WRITE CHECK INTERRUPT
1441                                    *******
1442
1443                                            THIS TEST WILL CHECK THAT A DCK DURING A WRITE CHECK WILL CAUSE
1444                                            AN INTERRUPT TO OCCUR.
```

```
1446
1447                    TEST 38 - CHECK ZERO FILL ON WRITE WITH WRITE CHECK
1448                    *******
1449
1450                            THIS TEST WILL VERIFY THAT WE CAN SUCCESSFULLY WRITE CHECK ALL
1451                            WORD COUNTS FROM 1 - 127.
1452
1453
1454                    TEST 39 - EXTENDED CHECK OF WRITE CHECK
1455                    *******
1456
1457                            THIS TEST WILL VERIFY THAT WE CAN WRITE CHECK SUCCESSFULLY
1458                            ALL PATTERNS.  PATTERNS USED ARE WALKING 1'S, 0'S, GROWING 1'S, 0'S.
1459
1460
1461                    TEST 40 - READ WITHOUT HEADER COMPARE
1462                    *******
1463
1464                            THIS TEST VERIFIES THAT THE FUNCTION READ WITHOUT HEADER COMPARE
1465                            (7) RESETS THE CONTROLLER READY AND POSTS NO ERRORS.  THE DISK
1466                            ADDRESS IS SET TO ALL ONES.
1467
1468
1469                    TEST 41 - READ WITHOUT HEADER COMPARE INTERRUPT
1470                    *******
1471
1472                            THIS TEST WILL VERIFY THAT THE FUNCTION READ WITHOUT HEADER
1473                            COMPARE (7) CAN GENERATE AN INTERRUPT ON COMPLETION.
1474
1475
1476                    TEST 42 - CHECK RD W/O HDR CMP READS
1477                    *******
1478
1479                            THIS TEST CHECKS THAT THE FUNCTION CAN ACTUALLY RECOVER DATA.
1480                            WE WRITE A PATTERN IN MEMORY AND CHECK THAT THE FUNCTION CAN
1481                            OVERLAY IT WITH DATA.
1482
1483
1484                    TEST 43 - CHECK RLBA INCREMENT WITH RD W/O HDR CMP
1485                    *******
1486
1487                            THIS TEST CHECKS THAT THE RLBA CAN INCREMENT PROPERLY ON THE
1488                            FUNCTION.
1489
1490
1491                    TEST 44 - CHECK RLDA DOES INCREMENT
1492                    *******
1493
1494                            THIS TEST CHECKS THAT THE RLDA DOES INCREMENT WITH THE
1495                            FUNCTION READ WITHOUT HEADER COMPARE.
1496
1497                    $
```

```
                                .ENABLE AMA
000000                          .ENABLE ABS
                                .MCALL SVC
        002000                  .=2000

                                .SBTTL  MACRO DEFINITIONS

                                .MACRO  CKERFG
                                        TST     ERFLG           ;ERROR IN HEADS HOME ROUTINE
                                        BEQ     123$            ;NO, THEN CONTINUE
                                        EXIT    TST             ;YES, EXIT TEST
                                123$:                           ;CONTINUE WITH TEST
                                .ENDM

                                .MACRO  WAITUS  ARG             ;MACRO MICRO-SEC WAIT
                                        MOV     ARG,XDELAY      ;SAVE ARGUMENT
                                        JSR     PC,TIME         ;CALL TIMING ROUTINE
                                .ENDM

                                .MACRO  WAITMS  ARG             ;MACRO MILLISEC WAIT
                                        MOV     ARG,YDELAY      ;SAVE ARGUMENT
                                        JSR     PC,XTIME        ;CALL TIMING ROUTINE
                                .ENDM

                                .NLIST  CND,MD,ME

002000                          SVC
        000000                  SVCINS=0
        000000                  SVCTAG=0

002000                                  POINTER BGNSW,BGNSFT,BGNDU

002000                          BGNMOD  MDHEDR

002000                                  HEADER  CNRLH,A,0,60,0,PRIO6            ;JSD REV A - ADDED PRIO6
002000      103                         .ASCII /C/
002001      116                         .ASCII /N/
002002      122                         .ASCII /R/
002003      114                         .ASCII /L/
002004      110                         .ASCII /H/
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   000060                         .WORD   60
002016   033562                         .WORD   L$HARD
002020   033736                         .WORD   L$SOFT
002022   012416                         .WORD   L$HW
002024   012434                         .WORD   L$SW
002026   034130                         .WORD   L$LAST
002030   000000                         .WORD   0
002032   000000                         .WORD   0
002034   000000                         .WORD   0
002036   000000                         .WORD   0
002040   012450                         .WORD   L$DISPATCH
```

MACRO DEFINITIONS

```
        002042  000300                          .WORD   PRIO6
        002044  000000                          .WORD   0
        002046  000000                          .WORD   0
        002050     003                          .BYTE   C$REVISION
        002051     003                          .BYTE   C$EDIT
        002052  000000                          .WORD   0
        002054  000000                          .WORD   0
        002056  000000                          .WORD   0
        002060  002220                          .WORD   L$DVTYP
        002062  000000                          .WORD   0
        002064  000000                          .WORD   0
        002066  000000                          .WORD   0
        002070  000000                          .WORD   0
        002072  013610                          .WORD   L$DU
        002074  000000                          .WORD   0
        002076  002122                          .WORD   L$DESC
        002100  104035                          EMT     E$LOAD
        002102  000000                          .WORD   0
        002104  012600                          .WORD   L$INIT
        002106  013514                          .WORD   L$CLEAN
        002110  013326                          .WORD   L$AUTO
        002112  012406                          .WORD   L$PROT
        002114  000000                          .WORD   0
        002116  000000                          .WORD   0
        002120  000000                          .WORD   0
  1534
  1535  002122                        ENDMOD
  1536
  1537  002122                                DESCRIPT        <CNRLH TESTS WRITE DATA, READ DATA, AND WRITE CHECK OPERATIONS>
        002122  103     116     122           .ASCIZ  /CNRLH TESTS WRITE DATA, READ DATA, AND WRITE CHECK OPERATIONS/
        002125  114     110     040
        002130  124     105     123
        002133  124     123     040
        002136  127     122     111
        002141  124     105     040
        002144  104     101     124
        002147  101     054     040
        002152  122     105     101
        002155  104     040     104
        002160  101     124     101
        002163  054     040     101
        002166  116     104     040
        002171  127     122     111
        002174  124     105     040
        002177  103     110     105
        002202  103     113     040
        002205  117     120     105
        002210  122     101     124
        002213  111     117     116
        002216  123     000
                                              .EVEN
  1538
  1539  002220                                DEVTYP  <RL01,RL02>
        002220  122     114     060           .ASCIZ  %RL01,RL02%
        002223  061     054     122
        002226  114     060     062
        002231  000
```

MACRO DEFINITIONS

```
                                        .EVEN

    1540
    1541                                .SBTTL  GLOBAL EQUATES
    1542
    1543 002232                         BGNMOD  GLBEQAT
    1544 002232                                 EQUALS
                                        ;
                                        ; BIT DIFINITIONS
                                        ;
                100000                  BIT15== 100000
                040000                  BIT14== 40000
                020000                  BIT13== 20000
                010000                  BIT12== 10000
                004000                  BIT11== 4000
                002000                  BIT10== 2000
                001000                  BIT09== 1000
                000400                  BIT08== 400
                000200                  BIT07== 200
                000100                  BIT06== 100
                000040                  BIT05== 40
                000020                  BIT04== 20
                000010                  BIT03== 10
                000004                  BIT02== 4
                000002                  BIT01== 2
                000001                  BIT00== 1
                                        ;
                001000                  BIT9==  BIT09
                000400                  BIT8==  BIT08
                000200                  BIT7==  BIT07
                000100                  BIT6==  BIT06
                000040                  BIT5==  BIT05
                000020                  BIT4==  BIT04
                000010                  BIT3==  BIT03
                000004                  BIT2==  BIT02
                000002                  BIT1==  BIT01
                000001                  BIT0==  BIT00
                                        ;
                                        ; EVENT FLAG DEFINITIONS
                                        ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                        ;
                                                                ;  BIT POSITION IN SECOND STATUS WORD
                000040                  EF.START==      32.     ; (100000) START COMMAND WAS ISSUED
                000037                  EF.RESTART==    31.     ; (040000) RESTART COMMAND WAS ISSUED
                000036                  EF.CONTINUE==   30.     ; (020000) CONTINUE COMMAND WAS ISSUED
                000035                  EF.NEW==        29.     ; (010000) A NEW PASS HAS BEEN STARTED
                000034                  EF.PWR==        28.     ; (004000) A POWER-FAIL/POWER-UP OCCURRED
                                        ;
                                        ;
                                        ; PRIORITY LEVEL DEFINITIONS
                                        ;
                000340                  PRI07== 340
                000300                  PRI06== 300
                000240                  PRI05== 240
                000200                  PRI04== 200
                000140                  PRI03== 140
                000100                  PRI02== 100
                000040                  PRI01== 40
```

GLOBAL EQUATES

```
                000000          PRIOO== 0
                                ;
                                ;OPERATOR FLAG BITS
                                ;
                000004          EVL==         4
                000010          LOT==        10
                000020          ADR==        20
                000040          IDU==        40
                000100          ISR==       100
                000200          UAM==       200
                000400          BOE==       400
                001000          PNT==      1000
                002000          PRI==      2000
                004000          IXE==      4000
                010000          IBE==     10000
                020000          IER==     20000
                040000          LOE==     40000
                100000          HOE==    100000
  1545          000001          DRDY=BIT0               ;DRIVE READY (RLCS)
  1546          000100          INTEN=BIT6              ;INTERRUPT ENABLE (RLCS)
  1547          100000          ERR=BIT15               ;RL11 ERROR (RLCS)
  1548          040000          DERR=BIT14              ;RL01 DRIVE ERROR (RLCS)
  1549          002000          OPI=BIT10               ;OPERATION INCOMPLETE (RLCS)
  1550          000200          CRDY=BIT7               ;CONTROLLER READY (RLCS)
  1551          000040          BA17=BIT5               ;EXTENDED ADDRESS BIT 17 (RLCS)
  1552          000020          BA16=BIT4               ;EXTENDED ADDRESS BIT 16 (RLCS)
  1553          020000          NXM=BIT13               ;NON-EXISTANT MEMORY (RLCS)
  1554          000000          DS0=0                   ;DRIVE SELECT 0 (RLCS)
  1555          000400          DS1=BIT8                ;DRIVE SELECT 1 (RLCS)
  1556          001000          DS2=BIT9                ;DRIVE SELECT 2 (RLCS)
  1557          001400          DS3=BIT8!BIT9           ;DRIVE SELECT 3 (RLCS)
  1558          000000          NOOP0=0                 ;FUNCTION-NOOP(0)
  1559          000002          WRCHK=BIT1              ;WRITE CHECK FUNCTION
  1560          000004          GSTAT=BIT2              ;GET STATUS FUNCTION
  1561          000006          SEEK=BIT2!BIT1          ;SEEK FUNCTION
  1562          000010          RDHDR=BIT3              ;READ HEADER FUNCTION
  1563          000012          WRITE=BIT3!BIT1         ;WRITE DATA FUNCTION
  1564          000014          READ=BIT3!BIT2          ;READ DATA FUNCTION
  1565          000016          RDNHD=BIT3!BIT2!BIT1    ;READ W/O HEADER VERIFICATION
  1566          000202          GODRVR=BIT1!BIT7        ;CRDY AND DRDY
  1567          000010          DRST=BIT3               ;DRIVE RESET (RLDA)
  1568          000002          GSBIT=BIT1              ;GET STATUS BIT (RLDA)
  1569          000001          MK=BIT0                 ;MARKER BIT (RLDA)
  1570          000004          SIGN=BIT2               ;SIGN BIT (RLDA)
  1571          000100          RHHS=BIT6               ;HEAD SELECT IN READ HEADER
  1572          000100          STHS=BIT6               ;HEAD SELECT IN STATUS BACK
  1573          000020          DAHS=BIT4               ;HEAD SELECT IN SEEK
  1574
  1575                          ;OFFSET FOR HARDWARE P-TABLE
  1576
  1577          000000          CSR=0
  1578          000002          VECT=2
  1579          000004          PRIOR=4
  1580          000006          TYPDR=6
  1581          000010          DRBT=10
  1582          000012          CNT=12
  1583
```

GLOBAL EQUATES

```
1584                                    ;OFFSET FOR SOFTWARE P-TABLE
1585
1586            000000                  DLT=0
1587            000002                  ELT=2
1588            000004                  SIZE=4
1589            000006                  DMPCK=6
1590            000010                  DLMT=10
1591            000012                  ANS=12
1592
1593 002232                             ENDMOD
1594
1595                                    .SBTTL  GLOBAL DATA
1596
1597 002232                             BGNMOD  GLBDAT
1598
1599 002232  000000                     T.DRIVE: .WORD  0
1600 002234  000000                     CHECK:   .WORD  0
1601 002236  000000                     T.CRC:   .WORD  0
1602 002240  000000                     WHY:     .WORD  0
1603 002242  000000                     CDCNT:   .WORD  0
1604 002244  000004                     ERRVEC:  .WORD  4
1605 002246  000000                     DRIVE:   .WORD  0
1606 002250  000000                     UUT:     .WORD  0
1607 002252  000000                     UNITST:  .WORD  0
1608 002254  000000                     TRPFLG:  .WORD  0
1609 002256  000000                     INTFLG:  .WORD  0               ;INTERRUPT OCCURANCE FLAG
1610 002260  000000                     LDCSR:   .WORD  0               ;LOCATION TO FORM RLCS
1611 002262  000077                     SECMSK:  .WORD  77              ;MASK OUT SECTOR
1612 002264  120001                     XPOLY:   .WORD  120001          ;POLYNOMIAL FOR CRC 16
1613 002266  000000                     BCCFBK:  .WORD  0               ;LOCATION USED BY "SIMBCC"
1614 002270  000000                     CALBCC:  .WORD  0               ;LOCATION USED BY "SIMBCC"
1615 002272  000000                     TMP0:    .WORD  0
1616 002274  000000                     TMP1:    .WORD  0
1617 002276  000000                     TMP2:    .WORD  0
1618 002300  000000                     GDDAT:   .WORD  0
1619 002302  000000                     BDDAT:   .WORD  0
1620 002304  000000                     TEMP2:   .WORD  0               'OCATION USED BY "SIMBCC"
1621 002306  000000                     TEMP3:   .WORD  0               ;LOCATION USED BY "SIMBCC"
1622 002310  000000                     TEMP4:   .WORD  0               ;LOCATION USED BY "SIMBCC"
1623 002312  000000                     FIRST:   .WORD  0               ;FIRST SECTOR READ
1624 002314  177700                     CYLMSK:  .WORD  177700          ;MASK CYLINDER AND HEAD SELECT
1625 002316  000050                     MXSEC1:  .WORD  40.             ;MAX SECTOR ADDRESS +1
1626 002320  000047                     MAXSEC:  .WORD  39.             ;MAX SECTOR ADDRESS
1627 002322  000000                     DWORD:   .WORD  0               ;DIFFERENCE WORD (SEEK)
1628 002324  177600                     MAXCYL:  .WORD  177600          ;MAXIMUM CYLINDER ADDRESS
1629 002326  000000                     SVHD:    .WORD  0               ;SAVE CURRENT HEAD SELECT
1630 002330  000000                     B.CS:    .WORD  0               ;CS - BEFORE OPERATION
1631 002332  000000                     B.BA:    .WORD  0               ;BA - BEFORE OPERATION
1632 002334  000000                     B.DA:    .WORD  0               ;DA - BEFORE OPERATION
1633 002336  000000                     B.MP:    .WORD  0               ;MP - BEFORE OPERATION
1634 002340  000000                     E.CS:    .WORD  0               ;CS - AT OCCURANCE OF ERROR
1635 002342  000000                     E.BA:    .WORD  0               ;BA - AT OCCURANCE OF ERROR
1636 002344  000000                     E.DA:    .WORD  0               ;DA - AT OCCURANCE OF ERROR
1637 002346  000000                     E.MP:    .WORD  0               ;MP - AT OCCURANCE OF ERROR
1638 002350  000000                     E.MP1:   .WORD  0
1639 002352  000000                     E.MP2:   .WORD  0
1640 002354  000000                     RLCS:    .WORD  0
```

GLOBAL DATA

```
1641 002356  000000                RLBA:    .WORD   0
1642 002360  000000                RLDA:    .WORD   0
1643 002362  000000                RLMP:    .WORD   0
1644 002364  000000                BCSR:    .WORD   0              ;CSR FROM P-TABLE
1645 002366  000000                BVEC:    .WORD   0              ;VECTOR FROM P-TABLE
1646 002370  000000                BPRIOR:  .WORD   0              ;BR LEVEL FROM P-TABLE
1647 002372  000000                FNDFNC:  .WORD   0
1648 002374  000000                XMEM:    .WORD   0
1649 002376  000000                TRYFNC:  .WORD   0              ;
1650 002400  000000                ERFLG:   .WORD   0
1651 002402  001212                LOPIMX:  .WORD   650.
1652 002404  000233                LOPIMN:  .WORD   155.
1653 002406  000620                UOPIMX:  .WORD   400.
1654 002410  000240                UOPIMN:  .WORD   160.
1655 002412  000000                OPIMN:   .WORD   0
1656 002414  000000                OPIMX:   .WORD   0
1657 002416  000000                PWRFLG:  .WORD   0
1658 002420  000000                T.CNTLR: .WORD   0
1659 002422  000000                DERFLG:  .WORD   0
1660 002424  000000                ERPOINT: .WORD   0
1661 002426                        ERCOUNT: .BLKW   64.
1662 002626  000000                XDELAY:  .WORD   0
1663 002630  000000                YDELAY:  .WORD   0
1664 002632  000000                TEMPO:   .WORD   0
1665 002634  000000                TEMP:    .WORD   0
1666 002636  000000                TIM.US:  .WORD   0
1667 002640  000000                TAG:     .WORD   0
1668 002642  000000                PCLKCS:  .WORD   0
1669 002644  000000                PCSR:    .WORD   0
1670 002646  000000                VEC:     .WORD   0
1671 002650  000000                HZ:      .WORD   0
1672 002652  000000                XITFLG:  .WORD   0
1673 002654  000000                FIFTY:   .WORD   0
1674 002656  000000                SIXTY:   .WORD   0
1675 002660  000000                PCLOCK:  .WORD   0
1676 002662  000000                NOTST:   .WORD   0
1677 002664  000000                OPITIM:  .WORD   0
1678 002666  000000                CLKFLD:  .WORD   0              ;CLOCK FIELD USED TO CHECK IF LSI-11 CLOCK
1679                                                               ;/IS "TICKING"
1680
1681                               .SBTTL   LIST TO CHECK HEADER COMPARE LOGIC
1682 002670  000000                HDRTAB:  .WORD   0              ;WALK 1
1683 002672  000001                         .WORD   BIT0
1684 002674  000002                         .WORD   BIT1
1685 002676  000004                         .WORD   BIT2
1686 002700  000010                         .WORD   BIT3
1687 002702  000020                         .WORD   BIT4
1688 002704  000040                         .WORD   BIT5
1689 002706  000100                         .WORD   BIT6
1690 002710  000200                         .WORD   BIT7
1691 002712  000400                         .WORD   BIT8
1692 002714  001000                         .WORD   BIT9
1693 002716  002000                         .WORD   BIT10
1694 002720  004000                         .WORD   BIT11
1695 002722  010000                         .WORD   BIT12
1696 002724  020000                         .WORD   BIT13
1697 002726  040000                         .WORD   BIT14
```

LIST TO CHECK HEADER COMPARE LOGIC

```
1698 002730  000003                        .WORD   3               ;GROW 1
1699 002732  000007                        .WORD   7
1700 002734  000017                        .WORD   17
1701 002736  000037                        .WORD   37
1702 002740  000137                        .WORD   137
1703 002742  000337                        .WORD   337
1704 002744  000737                        .WORD   737
1705 002746  001737                        .WORD   1737
1706 002750  003737                        .WORD   3737
1707 002752  007737                        .WORD   7737
1708 002754  017737                        .WORD   17737
1709 002756  037737                        .WORD   37737
1710 002760  077737                        .WORD   77737
1711 002762  077736                        .WORD   77736           ;GROW 0
1712 002764  077734                        .WORD   77734
1713 002766  077730                        .WORD   77730
1714 002770  077720                        .WORD   77720
1715 002772  077700                        .WORD   77700
1716 002774  077600                        .WORD   77600
1717 002776  077400                        .WORD   77400
1718 003000  077000                        .WORD   77000
1719 003002  076000                        .WORD   76000
1720 003004  074000                        .WORD   74000
1721 003006  070000                        .WORD   70000
1722 003010  060000                        .WORD   60000
1723 003012  040000                        .WORD   40000
1724 003014  077735                        .WORD   77735           ;WALK 0
1725 003016  077733                        .WORD   77733
1726 003020  077727                        .WORD   77727
1727 003022  077717                        .WORD   77717
1728 003024  077637                        .WORD   77637
1729 003026  077537                        .WORD   77537
1730 003030  077337                        .WORD   77337
1731 003032  076737                        .WORD   76737
1732 003034  075737                        .WORD   75737
1733 003036  073737                        .WORD   73737
1734 003040  067737                        .WORD   67737
1735 003042  057737                        .WORD   57737
1736 003044  037737                        .WORD   37737
1737 003046  000000              HDREND:   .WORD   0
1738 003050  000000              HTAB:     .WORD   0               ;WALK 1
1739 003052  000001                        .WORD   BIT0
1740 003054  000002                        .WORD   BIT1
1741 003056  000004                        .WORD   BIT2
1742 003060  000010                        .WORD   BIT3
1743 003062  000020                        .WORD   BIT4
1744 003064  000040                        .WORD   BIT5
1745 003066  000100                        .WORD   BIT6
1746 003070  000200                        .WORD   BIT7
1747 003072  000400                        .WORD   BIT8
1748 003074  001000                        .WORD   BIT9
1749 003076  002000                        .WORD   BIT10
1750 003100  004000                        .WORD   BIT11
1751 003102  010000                        .WORD   BIT12
1752 003104  020000                        .WORD   BIT13
1753 003106  040000                        .WORD   BIT14
1754 003110  100000                        .WORD   BIT15
```

L3

LIST TO CHECK HEADER COMPARE LOGIC

```
1755 003112  000003                          .WORD   3               ;GROW 1
1756 003114  000007                          .WORD   7
1757 003116  000017                          .WORD   17
1758 003120  000037                          .WORD   37
1759 003122  000137                          .WORD   137
1760 003124  000337                          .WORD   337
1761 003126  000737                          .WORD   737
1762 003130  001737                          .WORD   1737
1763 003132  003737                          .WORD   3737
1764 003134  007737                          .WORD   7737
1765 003136  017737                          .WORD   17737
1766 003140  037737                          .WORD   37737
1767 003142  077737                          .WORD   77737
1768 003144  177737                          .WORD   177737
1769 003146  177736                          .WORD   177736          ;GROW 0
1770 003150  177734                          .WORD   177734
1771 003152  177730                          .WORD   177730
1772 003154  177720                          .WORD   177720
1773 003156  177700                          .WORD   177700
1774 003160  177600                          .WORD   177600
1775 003162  177400                          .WORD   177400
1776 003164  177000                          .WORD   177000
1777 003166  176000                          .WORD   176000
1778 003170  174000                          .WORD   174000
1779 003172  170000                          .WORD   170000
1780 003174  160000                          .WORD   160000
1781 003176  140000                          .WORD   140000
1782 003200  100000                          .WORD   100000
1783 003202  177735                          .WORD   177735          ;WALK 0
1784 003204  177733                          .WORD   177733
1785 003206  177727                          .WORD   177727
1786 003210  177717                          .WORD   177717
1787 003212  177637                          .WORD   177637
1788 003214  177537                          .WORD   177537
1789 003216  177337                          .WORD   177337
1790 003220  176737                          .WORD   176737
1791 003222  175737                          .WORD   175737
1792 003224  173737                          .WORD   173737
1793 003226  167737                          .WORD   167737
1794 003230  157737                          .WORD   157737
1795 003232  137737                          .WORD   137737
1796 003234  000000                  HEND:   .WORD   0
1797
1798 003236  000001  000002  000004  DATPAT: .WORD   1,2,4,10,20,40,100,200,400,1000,2000,4000,10000,20000,40000,100000
     003244  000010  000020  000040
     003252  000100  000200  000400
     003260  001000  002000  004000
     003266  010000  020000  040000
     003274  100000
1799 003276  177777  177776  177775          .WORD   177777,177776,177775,177773,177767,177757,177737,177677
     003304  177773  177767  177757
     003312  177737  177677
1800 003316  177577  177377  176777          .WORD   177577,177377,176777,175777,173777,167777,157777,137777
     003324  175777  173777  167777
     003332  157777  137777
1801 003336  077777  177774  177770          .WORD   77777,177774,177770,177760,177740,177700,177600,177400
     003344  177760  177740  177700
```

LIST TO CHECK HEADER COMPARE LOGIC

```
                003352  177600  177400
1802            003356  177000  176000  174000          .WORD   177000,176000,174000,170000,160000,140000,3,7,17,37,77
                003364  170000  160000  140000
                003372  000003  000007  000017
                003400  000037  000077
1803            003404  000177  000377  000777          .WORD   177,377,777,1777,3777,7777,17777,37777,0
                003412  001777  003777  007777
                003420  017777  037777  000000
1804
1805            003426  000400                  BUF:    256.                    ;BUFFER FOR READ/WRITE
1806
1807            003430                          ENDMOD
1808
1809                                            .SBTTL  GLOBAL TEXT
1810
1811            003430                          BGNMOD  GLBTXT
1815            003430      103     123     072 ARLCS:  .ASCIZ  /CS: /
1816            003435      040     102     101 ARLBA:  .ASCIZ  / BA: /
1817            003443      040     104     101 ARLDA:  .ASCIZ  / DA: /
1818            003451      040     115     120 ARLMP:  .ASCIZ  / MP: /
1819            003457      102     105     106 BEREG:  .ASCIZ  /BEFORE COMMAND: /
1820            003500      124     111     115 AFREG:  .ASCIZ  /TIME OF ERROR:  /
1821            003521      103     117     116 CRTIM:  .ASCIZ  /CONTROLLER TIMED OUT/
1822            003546      104     122     111 DRTIM:  .ASCIZ  /DRIVE READY TIMED OUT/
1823            003574      040     104     122 DEMES:  .ASCIZ  / DRV/
1824            003601      040     116     130 NXMMES: .ASCIZ  / NXM/
1825            003606      040     117     120 OPIMES: .ASCIZ  / OPI/
1826            003613      040     110     103 HCRCMES:.ASCIZ  / HCRC/
1827            003621      040     110     116 HNFMES: .ASCIZ  / HNF/
1828            003626      040     104     103 DCKMES: .ASCIZ  / DCK/
1829            003633      040     104     114 DLTMES: .ASCIZ  / DLT/
1830            003640      015     000         LF:     .ASCIZ  <15>
1831            003642      015     012     000 MSCRLF: .ASCIZ  <15><12>
1832            003645      040     103     117 COMP:   .ASCIZ  / COMP/
1833            003653      106     122     103 OPIERR: .ASCIZ  /FRCD OPI C'SED OTHER ERRS/
1834            003705      116     117     117 NOPMES: .ASCIZ  /NOOP OPR'TN-FLAG MODE/
1835            003733      116     117     117 NOPINT: .ASCIZ  /NOOP OPR'TN-INTR. MODE/
1836            003762      127     122     111 WCKMES: .ASCIZ  /WRITE CHCK OPR'TN-FLAG MODE/
1837            004016      127     122     111 WCKINT: .ASCIZ  /WRITE CHCK OPR'TN-INTR. MODE/
1838            004053      122     104     040 RHDMES: .ASCIZ  /RD HDR OPR'TN-FLAG MODE/
1839            004103      122     104     040 RHDINT: .ASCIZ  /RD HDR OP-INTR. MODE/
1840            004130      123     113     040 SEKMES: .ASCIZ  /SK OP-FLAG MODE/
1841            004150      123     113     040 SEKINT: .ASCIZ  /SK OP-INTR. MODE/
1842            004171      107     105     124 GSTMES: .ASCIZ  /GET STATUS OP-FLAG MODE/
1843            004221      107     105     124 GSTINT: .ASCIZ  /GET STATUS OP-INTR MODE/
1844            004251      122     104     040 RDOMES: .ASCIZ  /RD OP-FLAG MODE/
1845            004271      122     104     040 RDOINT: .ASCIZ  /RD OP-INTR MODE/
1846            004311      127     122     124 WRTMES: .ASCIZ  /WRT OP-FLAG MODE/
1847            004332      127     122     124 WRTINT: .ASCIZ  /WRT OP-INTR MODE/
1848            004353      122     104     040 RDNMES: .ASCIZ  %RD W/O HDR - FLG MODE%
1849            004401      122     104     040 RDNINT: .ASCIZ  %RD W/O HDR - INTR MODE%
1850            004430      103     101     116 SKHOME: .ASCIZ  /CAN'T SK TO TRK 0/
1851            004452      127     122     124 WRLOCK: .ASCIZ  /WRT LOCK ERR/
1852            004467      122     114     103 EM1:    .ASCIZ  /RLCS HAD FOLLOWING ERR(S):/
1853            004522                          EM100:  .BLKB   120.
1854            004712      116     117     040 EM4:    .ASCIZ  /NO INTRPT ON RD OP/
1855            004735      122     104     040 EM5:    .ASCIZ  /RD OP DID NOT WRT MEM/
```

GLOBAL TEXT

```
1856 004763     122     114     102  EM6:    .ASCIZ  /RLBA DID NOT INCR DURING RD/
1857 005017     123     105     103  EM7:    .ASCIZ  /SECTR DID NOT INCR PROPERLY AFTER RD/
1858 005064     110     104     122  EM10:   .ASCIZ  /HDR NOT FND COULD NOT BE FORCED/
1859 005124     127     122     117  EM11:   .ASCIZ  /WRONG CYL ON SK/
1860 005144     110     104     122  EM12:   .ASCIZ  /HDR NOT FND WOULD NOT SET/
1861 005176     104     122     126  EM13:   .ASCIZ  /DRV RDY WOULD NOT SET/
1862 005224     104     123     113  EM14:   .ASCIZ  /DSK ADDR INCORRECT AFTER MULTIPLE SCTR READ/
1863 005300     104     122     126  EM16:   .ASCIZ  /DRV ERR ON WRT OP/
1864 005322     116     117     040  EM17:   .ASCIZ  /NO INTRPT ON WRT OP/
1865 005346     122     114     102  EM20:   .ASCIZ  /RLBA DID NOT INCR PROPERLY DURING WRT/
1866 005414     123     103     124  EM21:   .ASCIZ  /SCTR DID NOT INCR PROPERLY AFTER WRT/
1867 005461     104     123     113  EM22:   .ASCIZ  /DSK ADDR (RLDA) INCORRECT AFT MUL'PLE SCTR WRT/
1868 005540     110     104     122  EM23:   .ASCIZ  /HDR NOT FND COULD NOT BE FORCED AT END OF TRK/
1869 005616     116     130     115  EM24:   .ASCIZ  /NXM MEM ERR COULD NOT BE FORCED/
1870 005656     104     101     124  EM25:   .ASCIZ  %DATA CMP ERR - RD/WRT ERR%
1871 005710     127     122     124  EM26:   .ASCIZ  /WRT OP MODIFIED MEM/
1872 005734     105     122     122  EM27:   .ASCIZ  /ERR ON PARTIAL SCTR WRT - ZERO FILL CHCK/
1873 006005     122     114     102  EM30:   .ASCIZ  /RLBA DID NOT INCR PROPERLY/
1874 006040     102     101     040  EM31:   .ASCIZ  /BA BIT 16 DID NOT SET ON INCR/
1875 006076     102     101     040  EM32:   .ASCIZ  /BA BIT 17 SET ON BA16 INCR TST/
1876 006135     122     114     102  EM33:   .ASCIZ  /RLBA DID NOT INCR WITH BA16/
1877 006171     102     101     040  EM34:   .ASCIZ  /BA BIT 17 DID NOT SET ON INCR/
1878 006227     102     101     040  EM35:   .ASCIZ  /BA BIT 16 DID NOT CLR ON INCR/
1879 006265     122     114     102  EM36:   .ASCIZ  /RLBA DID NOT INCR WITH BA17/
1880 006321     122     105     101  EM40:   .ASCIZ  /READ(FUNCTION 7) DID NOT INTRPT/
1881 006361     122     104     050  EM41:   .ASCIZ  /RD(FUNCTION 7) ERR - BAD DATA/
1882 006417     122     104     040  EM42:   .ASCIZ  /RD (FUNCTION 7) ERR AT END OF TRK/
1883 006461     116     117     040  EM43:   .ASCIZ  /NO INTRPT WITH HDR NT FND FORCED/
1884 006522     116     117     040  EM44:   .ASCIZ  /NO INTRPT WITH NXM FORCED/
1885 006554     105     122     122  EM45:   .ASCIZ  %ERR ON BIT BANG OF SILO%
1886 006604     123     111     114  EM47:   .ASCIZ  /SILO OP FAIL/
1887 006621     110     104     122  EM50:   .ASCIZ  /HDR CMP FAILURE - SECTOR/
1888 006652     122     104     040  EM55:   .ASCIZ  ?RD W/O HDR CMP OP DID NOT WRT MEMORY?
1889 006717     122     114     102  EM53:   .ASCIZ  ?RLBA D'NT INCR DURING RD W/O HDR CMP?
1890 006764     122     114     104  EM54:   .ASCIZ  ?RLDA DID NOT INCR AFTER RD W/O HDR CMP?
1891 007033     117     120     111  EM56:   .ASCIZ  /OPI TIMING ERR/
1892 007052     127     122     124  EM57:   .ASCIZ  /WRT CHCK NPR CAUSED BUS TRAP/
1893 007107     127     122     124  EM60:   .ASCIZ  /WRT CHCK DID NOT INTRPT/
1894 007137     122     114     102  EM61:   .ASCIZ  /RLBA DID NOT INCR PROPERLY DURING WRCHK/
1895 007207     122     114     104  EM62:   .ASCIZ  /RLDA DID NOT INCR DURING WRCHK/
1896 007246     122     114     104  EM63:   .ASCIZ  /RLDA DID NOT INCR AFT A MULT' SCTR WRT CHK/
1897 007321     127     122     124  EM64:   .ASCIZ  /WRT CHECK OF PARTIAL SCTR WRT FAIL/
1898 007364     103     101     116  EM65:   .ASCIZ  /CANNOT FORCE DCK ON WRT CHCK/
1899 007421     103     101     116  EM66:   .ASCIZ  /CANNOT FORCE INTERRUPT WITH DCK ON WRCHK/
1900 007472     127     122     124  EM70:   .ASCIZ  /WRT CHCK FAIL/
1901
1902                                 .EVEN
1903
1907
1908 007510                                 ENDMOD
1909
1910 007510                         BGNMOD  GLBERR
1911
1912                                 .SBTTL  GLOBAL ERRORS
1913 007510                         BGNMSG  ERRO
1914
1915 007510  004737  010522                 JSR     PC,LINE1
```

GLOBAL ERRORS

```
1916 007514  004737  010556                    JSR     PC,LINE2
1917
1918 007520  004537  014526                    JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
1919
1920 007524                                    ENDMSG
     007524                          L10000:
     007524  104423                            TRAP    C$MSG
1921
1922 007526                          BGNMSG    ERR1
1923
1924 007526  004737  010522                    JSR     PC,LINE1
1925
1926 007532  004537  014526                    JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
1927
1928 007536                                    ENDMSG
     007536                          L10001:
     007536  104423                            TRAP    C$MSG
1929
1930 007540                          BGNMSG    ERR2
1931
1932 007540  004737  010522                    JSR     PC,LINE1
1933 007544                                    PRINTB  #FRMT4,GDDAT,BDDAT
     007544  013746  002302                    MOV     BDDAT,-(SP)
     007550  013746  002300                    MOV     GDDAT,-(SP)
     007554  012746  011170                    MOV     #FRMT4,-(SP)
     007560  012746  000003                    MOV     #3,-(SP)
     007564  010600                            MOV     SP,R0
     007566  104414                            TRAP    C$PNTB
     007570  062706  000010                    ADD     #10,SP
1934
1935 007574  004537  014526                    JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
1936
1937 007600                                    ENDMSG
     007600                          L10002:
     007600  104423                            TRAP    C$MSG
1938
1939 007602                          BGNMSG    ERR3
1940
1941 007602  004737  010522                    JSR     PC,LINE1
1942 007606  004737  010556                    JSR     PC,LINE2
1943 007612                                    PRINTB  #FRMT5,TMP0,BDDAT,GDDAT
     007612  013746  002300                    MOV     GDDAT,-(SP)
     007616  013746  002302                    MOV     BDDAT,-(SP)
     007622  013746  002272                    MOV     TMP0,-(SP)
     007626  012746  011226                    MOV     #FRMT5,-(SP)
     007632  012746  000004                    MOV     #4,-(SP)
     007636  010600                            MOV     SP,R0
     007640  104414                            TRAP    C$PNTB
     007642  062706  000012                    ADD     #12,SP
1944
1945 007646  004537  014526                    JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
1946
1947 007652                                    ENDMSG
     007652                          L10003:
     007652  104423                            TRAP    C$MSG
1948
1949 007654                          BGNMSG    ERR4
```

GLOBAL ERRORS

```
1950
1951 007654  004737  010522                  JSR     PC,LINE1
1952 007660  004737  010556                  JSR     PC,LINE2
1953 007664                                  PRINTB  #FRMT4,GDDAT,BDDAT
     007664  013746  002302                  MOV     BDDAT,-(SP)
     007670  013746  002300                  MOV     GDDAT,-(SP)
     007674  012746  011170                  MOV     #FRMT4,-(SP)
     007700  012746  000003                  MOV     #3,-(SP)
     007704  010600                          MOV     SP,R0
     007706  104414                          TRAP    C$PNTB
     007710  062706  000010                  ADD     #10,SP
1954
1955 007714  004537  014526                  JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
1956
1957 007720                                  ENDMSG
     007720                          L10004:
     007720  104423                          TRAP    C$MSG
1958
1959 007722                          BGNMSG  ERR5
1960
1961 007722  004737  010522                  JSR     PC,LINE1
1962 007726                                  PRINTB  #FRMT3,RESTMS
     007726  013746  015036                  MOV     RESTMS,-(SP)
     007732  012746  011163                  MOV     #FRMT3,-(SP)
     007736  012746  000002                  MOV     #2,-(SP)
     007742  010600                          MOV     SP,R0
     007744  104414                          TRAP    C$PNTB
     007746  062706  000006                  ADD     #6,SP
1963
1964 007752  004537  014526                  JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
1965
1966 007756                                  ENDMSG
     007756                          L10005:
     007756  104423                          TRAP    C$MSG
1967
1968 007760                          BGNMSG  ERR6
1969
1970 007760  004737  010522                  JSR     PC,LINE1
1971 007764  004737  011000                  JSR     PC,LINE3
1972 007770  004737  010556                  JSR     PC,LINE2
1973
1974 007774                                  PRINTB  #FRMT99
     007774  012746  012106                  MOV     #FRMT99,-(SP)
     010000  012746  000001                  MOV     #1,-(SP)
     010004  010600                          MOV     SP,R0
     010006  104414                          TRAP    C$PNTB
     010010  062706  000004                  ADD     #4,SP
1975 010014  004537  014526                  JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
1976
1977 010020                                  ENDMSG
     010020                          L10006:
     010020  104423                          TRAP    C$MSG
1978
1979 010022                          BGNMSG  ERR7
1980
1981 010022  004537  014526                  JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
1982
```

GLOBAL ERRORS

```
                                        ENDMSG
                                L10007:
                                        TRAP    C$MSG
                                BGNMSG  ERR8
                                        JSR     PC,LINE1
                                        JSR     PC,LINE2
                                        PRINTB  #FRMT6,TMP1,GDDAT,BDDAT
                                        MOV     BDDAT,-(SP)
                                        MOV     GDDAT,-(SP)
                                        MOV     TMP1,-(SP)
                                        MOV     #FRMT6,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0
                                        TRAP    C$PNTB
                                        ADD     #12,SP

                                        JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT

                                        ENDMSG
                                L10010:
                                        TRAP    C$MSG

                                BGNMSG  ERR9

                                        JSR     PC,LINE1
                                        JSR     PC,LINE2
                                        PRINTB  #FRMT4,TMP0,R2
                                        MOV     R2,-(SP)
                                        MOV     TMP0,-(SP)
                                        MOV     #FRMT4,-(SP)
                                        MOV     #3,-(SP)
                                        MOV     SP,R0
                                        TRAP    C$PNTB
                                        ADD     #10,SP

                                        JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT

                                        ENDMSG
                                L10011:
                                        TRAP    C$MSG

                                BGNMSG  ERR10

                                        JSR     PC,LINE1
                                        JSR     PC,LINE2
                                        PRINTB  #FRMT7,TMP1,GDDAT,BDDAT
                                        MOV     BDDAT,-(SP)
                                        MOV     GDDAT,-(SP)
                                        MOV     TMP1,-(SP)
                                        MOV     #FRMT7,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0
                                        TRAP    C$PNTB
                                        ADD     #12,SP
```

GLOBAL ERRORS

```
2011 010212  004537  014526                JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
2012
2013 010216                                ENDMSG
     010216                          L10012:
     010216  104423                        TRAP    C$MSG
2014
2015 010220                          BGNMSG  ERR11
2016
2017 010220  004737  010522                JSR     PC,LINE1
2018 010224  004737  010556                JSR     PC,LINE2
2019 010230                                PRINTB  #FRMT8,TMP0,GDDAT,BDDAT
     010230  013746  002302                MOV     BDDAT,-(SP)
     010234  013746  002300                MOV     GDDAT,-(SP)
     010240  013746  002272                MOV     TMP0,-(SP)
     010244  012746  011426                MOV     #FRMT8,-(SP)
     010250  012746  000004                MOV     #4,-(SP)
     010254  010600                        MOV     SP,R0
     010256  104414                        TRAP    C$PNTB
     010260  062706  000012                ADD     #12,SP
2020
2021 010264  004537  014526                JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
2022
2023 010270                                ENDMSG
     010270                          L10013:
     010270  104423                        TRAP    C$MSG
2024
2025 010272                          BGNMSG  ERR12
2026
2027 010272  004737  010522                JSR     PC,LINE1
2028 010276  004737  010556                JSR     PC,LINE2
2029 010302                                PRINTB  #FRMT9,TMP1,R3,GDDAT,BDDAT
     010302  013746  002302                MOV     BDDAT,-(SP)
     010306  013746  002300                MOV     GDDAT,-(SP)
     010312  010346                        MOV     R3,-(SP)
     010314  013746  002274                MOV     TMP1,-(SP)
     010320  012746  011547                MOV     #FRMT9,-(SP)
     010324  012746  000005                MOV     #5,-(SP)
     010330  010600                        MOV     SP,R0
     010332  104414                        TRAP    C$PNTB
     010334  062706  000014                ADD     #14,SP
2030
2031 010340  004537  014526                JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
2032
2033 010344                                ENDMSG
     010344                          L10014:
     010344  104423                        TRAP    C$MSG
2034
2035 010346                          BGNMSG  ERR13
2036
2037 010346  004737  010522                JSR     PC,LINE1
2038 010352                                PRINTB  #FRMT10,OPIMN,OPIMX,BDDAT
     010352  013746  002302                MOV     BDDAT,-(SP)
     010356  013746  002414                MOV     OPIMX,-(SP)
     010362  013746  002412                MOV     OPIMN,-(SP)
     010366  012746  011652                MOV     #FRMT10,-(SP)
     010372  012746  000004                MOV     #4,-(SP)
     010376  010600                        MOV     SP,R0
```

```
GLOBAL ERRORS

        010400  104414                          TRAP    C$PNTB
        010402  062706  000012                  ADD     #12,SP
   2039
   2040 010406  004537  014526                  JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
   2041
   2042 010412                                  ENDMSG
        010412                          L10015:
        010412  104423                          TRAP    C$MSG
   2043
   2044 010414                          BGNMSG  ERR14
   2045
   2046 010414  004737  010522                  JSR     PC,LINE1
   2047 010420  004737  010556                  JSR     PC,LINE2
   2048 010424                                  PRINTB  #FRMT14,TMP1,#BUF
        010424  012746  003426                  MOV     #BUF,-(SP)
        010430  013746  002274                  MOV     TMP1,-(SP)
        010434  012746  011476                  MOV     #FRMT14,-(SP)
        010440  012746  000003                  MOV     #3,-(SP)
        010444  010600                          MOV     SP,R0
        010446  104414                          TRAP    C$PNTB
        010450  062706  000010                  ADD     #10,SP
   2049
   2050 010454  004537  014526                  JSR     R5,CKERLT               ;INCREMENT ERROR AND CHECK LIMIT
   2051
   2052 010460                                  ENDMSG
        010460                          L10016:
        010460  104423                          TRAP    C$MSG
   2053
   2054 010462                          BGNMSG  ERR15
   2055
   2056 010462  004737  010522                  JSR     PC,LINE1
   2057 010466  004737  010556                  JSR     PC,LINE2
   2058 010472                                  PRINTB  #FRMT15,R2
        010472  010246                          MOV     R2,-(SP)
        010474  012746  012142                  MOV     #FRMT15,-(SP)
        010500  012746  000002                  MOV     #2,-(SP)
        010504  010600                          MOV     SP,R0
        010506  104414                          TRAP    C$PNTB
        010510  062706  000006                  ADD     #6,SP
   2059 010514  004537  014526                  JSR     R5,CKERLT
   2060
   2061 010520                                  ENDMSG
        010520                          L10017:
        010520  104423                          TRAP    C$MSG
   2062
   2063 010522                          LINE1:  PRINTB  #FRMT1,RLCS,<B,DRIVE+1>
        010522  005046                          CLR     -(SP)
        010524  153716  002247                  BISB    DRIVE+1,(SP)
        010530  013746  002354                  MOV     RLCS,-(SP)
        010534  012746  011052                  MOV     #FRMT1,-(SP)
        010540  012746  000003                  MOV     #3,-(SP)
        010544  010600                          MOV     SP,R0
        010546  104414                          TRAP    C$PNTB
        010550  062706  000010                  ADD     #10,SP
   2064 010554  000207                          RTS     PC
   2065
   2066 010556                          LINE2:  PRINTB  #FRMT2,#BEREG,#ARLCS,B.CS,#ARLBA,B.BA
```

GLOBAL ERRORS

```
        010556  013746  002332                  MOV     B.BA,-(SP)
        010562  012746  003435                  MOV     @ARLBA,-(SP)
        010566  013746  002330                  MOV     B.CS,-(SP)
        010572  012746  003430                  MOV     @ARLCS,-(SP)
        010576  012746  003457                  MOV     @BEREG,-(SP)
        010602  012746  011102                  MOV     @FRMT2,-(SP)
        010606  012746  000006                  MOV     #6,-(SP)
        010612  010600                          MOV     SP,R0
        010614  104414                          TRAP    C$PNTB
        010616  062706  000016                  ADD     #16,SP
  2067  010622                                  PRINTB  @FRMT2A,@ARLDA,B.DA,@ARLMP,B.MP
        010622  013746  002336                  MOV     B.MP,-(SP)
        010626  012746  003451                  MOV     @ARLMP,-(SP)
        010632  013746  002334                  MOV     B.DA,-(SP)
        010636  012746  003443                  MOV     @ARLDA,-(SP)
        010642  012746  011121                  MOV     @FRMT2A,-(SP)
        010646  012746  000005                  MOV     #5,-(SP)
        010652  010600                          MOV     SP,R0
        010654  104414                          TRAP    C$PNTB
        010656  062706  000014                  ADD     #14,SP
  2068  010662                                  PRINTB  @FRMT2,@AFREG,@ARLCS,E.CS,@ARLBA,E.BA
        010662  013746  002342                  MOV     E.BA,-(SP)
        010666  012746  003435                  MOV     @ARLBA,-(SP)
        010672  013746  002340                  MOV     E.CS,-(SP)
        010676  012746  003430                  MOV     @ARLCS,-(SP)
        010702  012746  003500                  MOV     @AFREG,-(SP)
        010706  012746  011102                  MOV     @FRMT2,-(SP)
        010712  012746  000006                  MOV     #6,-(SP)
        010716  010600                          MOV     SP,R0
        010720  104414                          TRAP    C$PNTB
        010722  062706  000016                  ADD     #16,SP
  2069  010726                                  PRINTB  @FRMT2B,@ARLDA,E.DA,@ARLMP,E.MP,E.MP1,E.MP2
        010726  013746  002352                  MOV     E.MP2,-(SP)
        010732  013746  002350                  MOV     E.MP1,-(SP)
        010736  013746  002346                  MOV     E.MP,-(SP)
        010742  012746  003451                  MOV     @ARLMP,-(SP)
        010746  013746  002344                  MOV     E.DA,-(SP)
        010752  012746  003443                  MOV     @ARLDA,-(SP)
        010756  012746  011134                  MOV     @FRMT2B,-(SP)
        010762  012746  000007                  MOV     #7,-(SP)
        010766  010600                          MOV     SP,R0
        010770  104414                          TRAP    C$PNTB
        010772  062706  000020                  ADD     #20,SP
  2070  010776  000207                          RTS     PC
  2071
  2072  011000                          LINE3:  PRINTB  @FRMT3,@EM1
        011000  012746  004467                  MOV     @EM1,-(SP)
        011004  012746  011163                  MOV     @FRMT3,-(SP)
        011010  012746  000002                  MOV     #2,-(SP)
        011014  010600                          MOV     SP,R0
        011016  104414                          TRAP    C$PNTB
        011020  062706  000006                  ADD     #6,SP
  2073  011024                                  PRINTB  @FRMT3,@EM100
        011024  012746  004522                  MOV     @EM100,-(SP)
        011030  012746  011163                  MOV     @FRMT3,-(SP)
        011034  012746  000002                  MOV     #2,-(SP)
        011040  010600                          MOV     SP,R0
```

GLOBAL ERRORS

```
        011042  104414                          TRAP    C$PNTB
        011044  062706  000006                  ADD     #6,SP
   2074 011050  000207                          RTS     PC
   2075
   2079
   2080 011052      045     101     103  FRMT1:  .ASCIZ  /%ACNTRLR: %O6%A DRV %O1/
   2081 011102      045     116     045  FRMT2:  .ASCIZ  /%N%T%T%O6%T%O6/
   2082 011121      045     124     045  FRMT2A: .ASCIZ  /%T%O6%T%O6/
   2083 011134      045     124     045  FRMT2B: .ASCIZ  /%T%O6%T%O6%A %O6%A %O6/
   2084 011163      045     116     045  FRMT3:  .ASCIZ  /%N%T/
   2085 011170      045     116     045  FRMT4:  .ASCIZ  /%N%AEXP'D: %O6%A REC'D: %O6%N/
   2086 011226      045     116     045  FRMT5:  .ASCIZ  /%N%ALAST: %O6%A PRES: %O6%A EXP'D: %O6%N/
   2087 011277      045     116     045  FRMT6:  .ASCIZ  /%N%ABUS ADR: %O6%A EXP'D: %O6%A REC'D: %O6%N/
   2088 011354      045     116     045  FRMT7:  .ASCIZ  /%N%AWORD: %D3%A EXP'D: %O6%A REC'D: %O6%N/
   2089 011426      045     116     045  FRMT8:  .ASCIZ  /%N%ADA: %O6%A REC'D: %O6%A EXP'D: %O6%N/
   2090 011476      045     116     045  FRMT14: .ASCIZ  /%N%AWORDS WRITTEN: %D3%A BUS ADDR: %O6%N/
   2091 011547      045     116     045  FRMT9:  .ASCIZ  /%N%AWORDS WRITTEN: %D3%A BUS ADDR: %O6%A EXP'D: %O6%A REC'D: %O6%N/
   2092 011652      045     116     045  FRMT10: .ASCII  /%N%ARANGE %D3%A - %D3%A  MILLISECONDS WAS %D6%N/
   2093 011731      045     101     115          .ASCIZ  /%AMAX TIMEOUT OF PROGRAM IS 3 SECONDS%N/
   2094 012001      045     116     045  FRMT11: .ASCIZ  /%N%AERR LIMIT EXCEEDED - DROPPED%N/
   2095 012044      045     101     104  FRMT98: .ASCII  /%ADRV DID NOT RCVR FROM POWER FAIL/
   2096 012106      045     116     000  FRMT99: .ASCIZ  /%N/
   2097 012111      045     116     045  FRMT13: .ASCIZ  /%N%T%A - WILL NOT TEST%N/
   2098 012142      045     116     045  FRMT15: .ASCIZ  /%N%APATTERN WAS: %O6/
   2099 012167      045     116     045  FRMT16: .ASCIZ  /%N%ADRIVE DROPPED - NO CONTROLLER%N/
   2100 012233      045     116     045  FRMT17: .ASCIZ  /%N%ADRIVE DROPPED - DID NOT RESPOND WITH "READY"%N/
   2101 012316      045     116     045  FRMT18: .ASCIZ  /%N%ATEST 7 CANNOT BE PERFORMED...CLOCK IS NOT AVAILABLE/
   2102
   2103                                  .EVEN
   2104
   2108
   2109 012406                           ENDMOD
   2110
   2111                                  ;LOAD PROTECTION TABLE
   2112 012406                           BGNPROT
   2113 012406  000000                          .WORD    0              ;OFFSET OF CSR IN P-TABLE
   2114 012410  177777                          .WORD   -1              ;NOT A MASS-BUS DRIVE
   2115 012412  000010                          .WORD   10              ;OFFSET OF DRIVE IN P-TABLE
   2116 012414                           ENDPROT
   2117
   2118 012414                           BGNMOD  HPTCODE
   2119 012414                           BGNHW
        012414  000006                          .WORD   L10021-L$HW/2
   2120
   2121 012416  174400                          .WORD   174400          ;CSR
   2122 012420  000160                          .WORD   160             ;VECTOR
   2123 012422  000240                          .WORD   240             ;PRIORITY
   2124 012424  000001                          .WORD   1               ;TYPE OF DRIVE RL01 OR RL02
   2125 012426  000000                          .WORD   0               ;DRIVE (BITS 8,9,10)
   2126 012430  000001                          .WORD   1               ;RL11=1 RLV11=0
   2127
   2128 012432                           ENDHW
        012432                           L10021:
   2129 012432                           ENDMOD
   2130
   2131 012432                           BGNMOD  SPTCODE
   2132 012432                           BGNSW
```

GLOBAL ERRORS

```
        012432  000005                          .WORD   L10022-L$SW/2
   2133
   2134 012434  000000          DROP:   .WORD   0
   2135 012436  000012          MERLMT: .WORD   10.
   2136 012440  000000          T.DMP:  .WORD   0
   2137 012442  000000          T.LMT:  .WORD   0
   2138 012444  000001          T.ANS:  .WORD   1
   2139
   2140 012446                  ENDSW
        012446                  L10022:
   2141 012446                  ENDMOD
   2142
   2143 012446                  BGNMOD  DSPCODE
   2144
   2145 012446                  DISPATCH        44
        012446  000054                  .WORD   44
        012450  016240                  .WORD   T1
        012452  016404                  .WORD   T2
        012454  016534                  .WORD   T3
        012456  016670                  .WORD   T4
        012460  017022                  .WORD   T5
        012462  017160                  .WORD   T6
        012464  017356                  .WORD   T7
        012466  017760                  .WORD   T8
        012470  020150                  .WORD   T9
        012472  020346                  .WORD   T10
        012474  020520                  .WORD   T11
        012476  020716                  .WORD   T12
        012500  021116                  .WORD   T13
        012502  021220                  .WORD   T14
        012504  021344                  .WORD   T15
        012506  021540                  .WORD   T16
        012510  021674                  .WORD   T17
        012512  022026                  .WORD   T18
        012514  022146                  .WORD   T19
        012516  022326                  .WORD   T20
        012520  023140                  .WORD   T21
        012522  023334                  .WORD   T22
        012524  023500                  .WORD   T23
        012526  023664                  .WORD   T24
        012530  024050                  .WORD   T25
        012532  024450                  .WORD   T26
        012534  025072                  .WORD   T27
        012536  025520                  .WORD   T28
        012540  026200                  .WORD   T29
        012542  026632                  .WORD   T30
        012544  027246                  .WORD   T31
        012546  027500                  .WORD   T32
        012550  027770                  .WORD   T33
        012552  030264                  .WORD   T34
        012554  030556                  .WORD   T35
        012556  031150                  .WORD   T36
        012560  031450                  .WORD   T37
        012562  032010                  .WORD   T38
        012564  032322                  .WORD   T39
        012566  032646                  .WORD   T40
        012570  032736                  .WORD   T41
```

GLOBAL ERRORS

```
        012572  033070                          .WORD   T42
        012574  033266                          .WORD   T43
        012576  033424                          .WORD   T44
  2146
  2147 012600                           ENDMOD
  2148
  2149                                  .SBTTL  INITIALIZATION CODE
  2150
  2151 012600                           BGNMOD  INITCODE
  2152 012600                           BGNINIT
  2153                                  ;       SETPRI  #PRIO7                          ;JSD REV A
  2154 012600                                   SETPRI  #PRIO6                          ;JSD REV A
       012600  012700  000300                   MOV     #PRIO6,R0
       012604  104441                           TRAP    C$SPRI
  2155 012606                                   READEF  #EF.PWR
       012606  012700  000034                   MOV     #EF.PWR,R0
       012612  104447                           TRAP    C$REFG
  2156 012614                                   BNCOMPLETE      NOPWR
       012614  103004                           BCC     NOPWR
  2157 012616  013737  002012  002416           MOV     L$UNIT,PWRFLG
  2158 012624  000510                           BR      CONT
  2159 012626                           NOPWR:  READEF  #EF.RESTART
       012626  012700  000037                   MOV     #EF.RESTART,R0
       012632  104447                           TRAP    C$REFG
  2160 012634                                   BCOMPLETE       START1
       012634  103404                           BCS     START1
  2161 012636                                   READEF  #EF.START
       012636  012700  000040                   MOV     #EF.START,R0
       012642  104447                           TRAP    C$REFG
  2162 012644                                   BNCOMPLETE      CONTINUET
       012644  103023                           BCC     CONTINUET
  2163 012646                           START1: SETVEC  #140,#170000,#340       ;ODT STARTING ADDR      ;JSD REV A
       012646  012746  000340                   MOV     #340,-(SP)
       012652  012746  170000                   MOV     #170000,-(SP)
       012656  012746  000140                   MOV     #140,-(SP)
       012662  012746  000003                   MOV     #3,-(SP)
       012666  104437                           TRAP    C$SVEC
       012670  062706  000010                   ADD     #10,SP
  2164 012674  012700  002426                   MOV     #ERCOUNT,R0
  2165 012700  012701  000100                   MOV     #64.,R1
  2166 012704  005020                   1$:     CLR     (R0)+
  2167 012706  005301                           DEC     R1
  2168 012710  001375                           BNE     1$
  2169 012712  000407                           BR      START
  2170 012714                           CONTINUE:       READEF  #EF.CONTINUE
       012714  012700  000036                   MOV     #EF.CONTINUE,R0
       012720  104447                           TRAP    C$REFG
  2171 012722                                   BCOMPLETE       CONT
       012722  103451                           BCS     CONT
  2172 012724  005737  002250           NXT:    TST     UUT                             ;DONE WITH ALL UNITS
  2173 012730  001011                           BNE     XXX                             ;NO
  2174 012732  012737  177777  002252   START:  MOV     #-1,UNITST
  2175 012740  013737  002012  002250           MOV     L$UNIT,UUT
  2176 012746  012737  002424  002424           MOV     #ERCOUNT-2,ERPOINT
  2177 012754  005237  002252           XXX:    INC     UNITST
  2178 012760  062737  000002  002424           ADD     #2,ERPOINT
  2179 012766  005337  002250                   DEC     UUT
```

INITIALIZATION CODE

```
2180 012772                          REST:   GPHARD  UNITST,R0
     012772  013700  002252                  MOV     UNITST,R0
     012776  104442                          TRAP    C$GPHRD
2181 013000                                  BCOMPLETE       2$
     013000  103406                          BCS     2$
2182 013002  005737  002416                  TST     PWRFLG
2183 013006  001746                          BEQ     NXT
2184 013010  005337  002416                  DEC     PWRFLG
2185 013014  000743                          BR      NXT
2186 013016  012037  002364          2$:     MOV     (R0)+,BCSR              ;GET BUS ADDRESS
2187 013022  012037  002366                  MOV     (R0)+,BVEC              ;GET VECTOR
2188 013026  012037  002370                  MOV     (R0)+,BPRIOR            ;GET PRIORITY
2189 013032  012037  002232                  MOV     (R0)+,T.DRIVE           ;GET TYPE OF DRIVE
2190 013036  012037  002246                  MOV     (R0)+,DRIVE             ;GET DRIVE
2191 013042  012037  002420                  MOV     (R0)+,T.CNTLR           ;GET CONTROLLER TYPE
2192 013046  013700  002364          CONT:   MOV     BCSR,R0                 ;CREATE REGISTERS
2193 013052  010037  002354                  MOV     R0,RLCS
2194 013056  062700  000002                  ADD     #2,R0
2195 013062  010037  002356                  MOV     R0,RLBA
2196 013066  062700  000002                  ADD     #2,R0
2197 013072  010037  002360                  MOV     R0,RLDA
2198 013076  062700  000002                  ADD     #2,R0
2199 013102  010037  002362                  MOV     R0,RLMP
2200 013106  005737  002416                  TST     PWRFLG                  ;POWER UP?
2201 013112  001452                          BEQ     END                     ;NO
2202 013114  012777  000200  167232          MOV     #200,@RLCS
2203 013122  053777  002246  167224          BIS     DRIVE,@RLCS
2204 013130  012701  000170                  MOV     #120.,R1                ;INITIALIZE WAIT COUNT
2205 013134                          3$:     WAITMS  #10.
2206 013146  032777  000001  167200          BIT     #1,@RLCS
2207 013154  001031                          BNE     END
2208 013156  005301                          DEC     R1
2209 013160  001365                          BNE     3$
2210 013162                                  PRINTF  #FRMT99
     013162  012746  012106                  MOV     #FRMT99,-(SP)
     013166  012746  000001                  MOV     #1,-(SP)
     013172  010600                          MOV     SP,R0
     013174  104417                          TRAP    C$PNTF
     013176  062706  000004                  ADD     #4,SP
2211 013202                                  PRINTF  #FRMT98
     013202  012746  012044                  MOV     #FRMT98,-(SP)
     013206  012746  000001                  MOV     #1,-(SP)
     013212  010600                          MOV     SP,R0
     013214  104417                          TRAP    C$PNTF
     013216  062706  000004                  ADD     #4,SP
2212 013222  004737  010522                  JSR     PC,LINE1
2213 013226                                  DODU    UNITST
     013226  013700  002252                  MOV     UNITST,R0
     013232  104451                          TRAP    C$DODU
2214 013234  000137  012724                  JMP     NXT
2215 013240  013737  002410  002412  END:    MOV     UOPIMN,OPIMN
2216 013246  013737  002406  002414          MOV     UOPIMX,OPIMX
2217 013254  005737  002420                  TST     T.CNTLR                 ;RL11??
2218 013260  001006                          BNE     1$                      ;YES, THEN KEEP LIMITS SET
2219 013262  013737  002404  002412          MOV     LOPIMN,OPIMN
2220 013270  013737  002402  002414          MOV     LOPIMX,OPIMX
2221 013276                          1$:     SETVEC  BVEC,#INTSRV,#340
```

```
INITIALIZATION CODE

        013276  012746  000340                  MOV     #340,-(SP)
        013302  012746  014464                  MOV     #INTSRV,-(SP)
        013306  013746  002366                  MOV     BVEC,-(SP)
        013312  012746  000003                  MOV     #3,-(SP)
        013316  104437                          TRAP    C$SVEC
        013320  062706  000010                  ADD     #10,SP
   2222 013324                          ENDINIT
        013324                          L10023:
        013324  104411                          TRAP    C$INIT
   2223 013326                          ENDMOD
   2224
   2225                                 .SBTTL  AUTO DROP SECTION
   2226
   2227 013326                          BGNAUTO
   2228 013326  005037  002254                  CLR     TRPFLG          ;CLEAR TRAP FLAG
   2229 013332                                  SETVEC  ERRVEC,#TRPHAN,#340 ;SET UP TRAP VECTOR TO DETECT
        013332  012746  000340                  MOV     #340,-(SP)
        013336  012746  015756                  MOV     #TRPHAN,-(SP)
        013342  013746  002244                  MOV     ERRVEC,-(SP)
        013346  012746  000003                  MOV     #3,-(SP)
        013352  104437                          TRAP    C$SVEC
        013354  062706  000010                  ADD     #10,SP
   2230                                                                 ;/NON-EXISTENT CONTROLLER
   2231 013360  005777  166770                  TST     @RLCS           ;ACCESS CONTROLLER
   2232 013364                                  CLRVEC  ERRVEC          ;RELEASE TRAP VECTOR
        013364  013700  002244                  MOV     ERRVEC,R0
        013370  104436                          TRAP    C$CVEC
   2233 013372  005737  002254                  TST     TRPFLG          ;DID IT TRAP?
   2234 013376  001416                          BEQ     1$              ;NO - CHECK ITS DRIVE
   2235 013400                                  PRINTB  #FRMT16         ;ELSE, PRINT MSG. "DRIVE DROPPED - NO CONTROLLER"
        013400  012746  012167                  MOV     #FRMT16,-(SP)
        013404  012746  000001                  MOV     #1,-(SP)
        013410  010600                          MOV     SP,R0
        013412  104414                          TRAP    C$PNTB
        013414  062706  000004                  ADD     #4,SP
   2236 013420  004737  010522                  JSR     PC,LINE1        ;PROVIDE DRIVE INFORMATION
   2237 013424                                  DODU    UNITST          ;DO DROP UNIT ON DRIVE
        013424  013700  002252                  MOV     UNITST,R0
        013430  104451                          TRAP    C$DODU
   2238 013432  000427                          BR      2$              ;EXIT
   2239 013434  012777  000200  166712  1$:     MOV     #200,@RLCS      ;SET CONTROLLER READY
   2240 013442  053777  002246  166704          BIS     DRIVE,@RLCS     ;SELECT DRIVE
   2241 013450  032777  000001  166676          BIT     #1,@RLCS        ;IS DRIVE READY?
   2242 013456  001015                          BNE     2$              ;YES - EXIT
   2243 013460                                  PRINTB  #FRMT17         ;ELSE, PRINT MSG. "DRIVE DROPPED - DID NOT
        013460  012746  012233                  MOV     #FRMT17,-(SP)
        013464  012746  000001                  MOV     #1,-(SP)
        013470  010600                          MOV     SP,R0
        013472  104414                          TRAP    C$PNTB
        013474  062706  000004                  ADD     #4,SP
   2244                                                                 ;/RESPOND WITH 'READY'"
   2245 013500  004737  010522                  JSR     PC,LINE1        ;PROVIDE DRIVE INFORMATION
   2246 013504                                  DODU    UNITST          ;DO DROP UNIT ON DRIVE
        013504  013700  002252                  MOV     UNITST,R0
        013510  104451                          TRAP    C$DODU
   2247 013512                          2$:
   2248 013512                          ENDAUTO
```

AUTO DROP SECTION

```
        013512
        013512  104461                          L10024:
                                                        TRAP    C$AUTO
  2249
  2250                                          .SBTTL  CLEANUP CODE SECTION
  2251
  2252 013514                                   BGNMOD  CLNCODE
  2253 013514                                   BGNCLN
  2254
  2255 013514                                           SETVEC  ERRVEC,@TRPHAN,@340
        013514  012746  000340                          MOV     @340,-(SP)
        013520  012746  015756                          MOV     @TRPHAN,-(SP)
        013524  013746  002244                          MOV     ERRVEC,-(SP)
        013530  012746  000003                          MOV     @3,-(SP)
        013534  104437                                  TRAP    C$SVEC
        013536  062706  000010                          ADD     @10,SP
  2256 013542  032777  000200  166604  1$:             BIT     @CRDY,@RLCS
  2257 013550  001774                                  BEQ     1$
  2258 013552  042777  000100  166574                  BIC     @INTEN,@RLCS
  2259 013560                                           CLRVEC  BVEC
        013560  013700  002366                          MOV     BVEC,R0
        013564  104436                                  TRAP    C$CVEC
  2260 013566  005737  002416                          TST     PWRFLG
  2261 013572  001402                                  BEQ     2$
  2262 013574  005337  002416                          DEC     PWRFLG
  2263 013600                                   2$:     CLRVEC  ERRVEC
        013600  013700  002244                          MOV     ERRVEC,R0
        013604  104436                                  TRAP    C$CVEC
  2264
  2265 013606                                   ENDCLN
        013606                                  L10025:
        013606  104412                                  TRAP    C$CLEAN
  2266 013610                                   ENDMOD
  2267
  2279
  2280                                          .SBTTL  GLOBAL SUBROUTINES
  2281
  2282 013614                                   BGNMOD  GLBSUB
  2283
  2284 013614  012737  000160  002116  TIME:    MOV     #160,L$DLY      ;GET OUTER DELAY LOOP
  2285 013622  005237  002636                    INC     TIM.US          ;US-WAIT ROUTINE INDICATOR
  2286 013626  005437  002626                    NEG     XDELAY          ;GET NEGATIVE OF FACTOR
  2287 013632  005737  002420                    TST     T.CNTLR         ;RL11?
  2288 013636  001420                            BEQ     2$              ;BRANCH - IF NO
  2289                                  ;1$:      DELAY   #1.             ;WAIT AT LEAST 100 US--         ;JSD REV A
  2290 013640                           1$:      DELAY   1.              ;WAIT AT LEAST 100 US--         ;JSD REV A
        013640  012727  000001                    MOV     #1.,(PC)+
        013644  000000                            .WORD   0
        013646  013727  002116                    MOV     L$DLY,(PC)+
        013652  000000                            .WORD   0
        013654  005367  177772                    DEC     -6(PC)
        013660  001375                            BNE     .-4
        013662  005367  177756                    DEC     -22(PC)
        013666  001367                            BNE     .-20
  2291 013670  005237  002626                    INC     XDELAY          ;WAIT FACTOR EXPIRED?
  2292 013674  002761                            BLT     1$              ;BRANCH - IF NO
  2293 013676  000422                            BR      4$              ;EXIT
  2294 013700  012737  000150  002116  2$:      MOV     #150,L$DLY      ;GET OUTER DELAY LOOP
```

GLOBAL SUBROUTINES

```
 2295                                   ;3$:    DELAY   #1.                     ;WAIT WITH RESPECT TO FONZ BUS  ;JSD REV A
 2296 013706                            3$:     DELAY   1.                      ;WAIT WITH RESPECT TO FONZ BUS  ;JSD REV A
      013706  012727  000001                   MOV     #1.,(PC)+
      013712  000000                           .WORD   0
      013714  013727  002116                   MOV     L$DLY,(PC)+
      013720  000000                           .WORD   0
      013722  005367  177772                   DEC     -6(PC)
      013726  001375                           BNE     .-4
      013730  005367  177756                   DEC     -22(PC)
      013734  001367                           BNE     .-20
 2297 013736  005237  002626                   INC     XDELAY                  ;WAIT FACTOR EXPIRED?
 2298 013742  002761                           BLT     3$                      ;BRANCH - IF NO
 2299 013744  000207                   4$:     RTS     PC                      ;RETURN
 2300
 2301 013746  012737  000160  002116 XTIME:  MOV     #160,L$DLY              ;GET OUTER DELAY LOOP
 2302 013754  005037  002636                   CLR     TIM.US                  ;MS WAIT INDICATOR
 2303 013760  006337  002630                   ASL     YDELAY                  ;MULTIPLY BY FACTOR 4
 2304 013764  006337  002630                   ASL     YDELAY                  ; -------
 2305 013770  005437  002630                   NEG     YDELAY                  ;GET NEGATIVE OF RESULT
 2306 013774  005737  002420                   TST     T.CNTLR                 ;RL11?
 2307 014000  001023                           BNE     1$                      ;BRANCH - IF YES
 2308 014002  012737  000150  002116           MOV     #150,L$DLY              ;GET OUTER DELAY LOOP
 2309                                   ;2$:    DELAY   #20                     ;WAIT WITH RESPECT TO FONZ BUS  ;JSD REV A
 2310 014010                            2$:     DELAY   20                      ;WAIT WITH RESPECT TO FONZ BUS  ;JSD REV A
      014010  012727  000020                   MOV     #20,(PC)+
      014014  000000                           .WORD   0
      014016  013727  002116                   MOV     L$DLY,(PC)+
      014022  000000                           .WORD   0
      014024  005367  177772                   DEC     -6(PC)
      014030  001375                           BNE     .-4
      014032  005367  177756                   DEC     -22(PC)
      014036  001367                           BNE     .-20
 2311 014040  005237  002630                   INC     YDELAY                  ;WAIT FACTOR EXPIRED?
 2312 014044  002761                           BLT     2$                      ;BRANCH - IF NO
 2313 014046  000417                           BR      3$                      ;GET TIME
 2314                                   ;1$:    DELAY   #10                     ;WAIT AT LEAST 25 MS            ;JSD REV A
 2315 014050                            1$:     DELAY   10                      ;WAIT AT LEAST 25 MS            ;JSD REV A
      014050  012727  000010                   MOV     #10,(PC)+
      014054  000000                           .WORD   0
      014056  013727  002116                   MOV     L$DLY,(PC)+
      014062  000000                           .WORD   0
      014064  005367  177772                   DEC     -6(PC)
      014070  001375                           BNE     .-4
      014072  005367  177756                   DEC     -22(PC)
      014076  001367                           BNE     .-20
 2316 014100  005237  002630                   INC     YDELAY                  ;WAIT FACTOR EXPIRED?
 2317 014104  002761                           BLT     1$                      ;BRANCH - IF NO
 2318 014106  000207                   3$:     RTS     PC                      ;RETURN
 2319
 2320 014110  010146                   SETCLK: MOV     R1,-(SP)                ;SAVE R1
 2321                                   ;                                                               ;JSD REV A
 2322                                   ;       UNDER FALCON-PLUS, CLOCK OPERATION IS NOT GUARANTEED.  CLOCK    ;JSD REV A
 2323                                   ;       INTERRUPTS MAY OR MAY NOT BE HARD-ENABLED, AND EVEN IF THEY WERE,  ;JSD REV A
 2324                                   ;       THE INTERRUPT RATE COULD BE 50, 60, OR 800 HERTZ.  FURTHERMORE, THE  ;JSD REV A
 2325                                   ;       FOLLOWING CLOCK MACROS PROVIDE MISLEADING INFORMATION (UNDER            ;JSD
REV A
 2326                                   ;       FALCON-PLUS).  FOR ALL THESE REASONS, ASSUME NO CLOCK PRESENT, AND       ;JSD
REV A
 2327                                   ;       DON'T RUN TEST 7.                                                       ;JSD
REV A
```

```
 2328                                   ;                                                                       ;JSD
REV A
 2329                                   ;       CLOCK   P,PCLKCS            ;PROGRAMMABLE CLOCK AVAILABLE? - CSR=772540 ;JSD
REV A
 2330                                   ;       BCOMPLETE       1$          ;BRANCH - IF YES                        ;JSD
REV A
 2331                                   ;       CLOCK   L,PCLKCS            ;LINE CLOCK AVAILABLE? - CSR=777546     ;JSD
REV A
 2332                                   ;       BCOMPLETE       20$         ;BRANCH IF L-CLOCK                      ;JSD
REV A
 2333 014112  000462                            BR      2$                  ;ELSE, INDICATE CLOCK IS NOT PRESENT
 2334 014114                            20$:    READBUS                     ;CHECK TYPE OF BUS
      014114  104407                            TRAP    C$RDBU
 2335 014116                                    BNCOMPLETE      1$          ;BRANCH IF NOT Q-BUS
      014116  103036                            BCC     1$
 2336 014120  005037  002666                    CLR     CLKFLD              ;CLEAR CLOCK FIELD FOR STORING "TICKS"
 2337 014124                                    SETVEC  #100,#CLKTIK,#340   ;SET UP LSI-11 L-CLOCK INTERRUPT VECTOR
      014124  012746  000340                    MOV     #340,-(SP)
      014130  012746  014520                    MOV     #CLKTIK,-(SP)
      014134  012746  000100                    MOV     #100,-(SP)
      014140  012746  000003                    MOV     #3,-(SP)
      014144  104437                            TRAP    C$SVEC
      014146  062706  000010                    ADD     #10,SP
 2338                                                                       ;/TO CHECK IF CLOCK IS "TICKING"
 2339 014152                                    SETPRI  #PRI05              ;SET PRIORITY TO 5 TO ALLOW CLOCK INTERRUPTS
      014152  012700  000240                    MOV     #PRI05,R0
      014156  104441                            TRAP    C$SPRI
 2340 014160                                    WAITMS  #5                  ;PAUSE TO ALLOW CLOCK INTERRUPTS
 2341                                   ;       SETPRI  #PRI07              ;RESTORE PRI TO 7 TO INHIBIT INT'S      ;JSD REV A
 2342 014172                                    SETPRI  #PRI06              ;RESTORE PRI TO 6 TO INHIBIT INT'S      ;JSD REV A
      014172  012700  000300                    MOV     #PRI06,R0
      014176  104441                            TRAP    C$SPRI
 2343 014200                                    CLRVEC  #100                ;CLEAR L-CLOCK INTERRUPT VECTOR
      014200  012700  000100                    MOV     #100,R0
      014204  104436                            TRAP    C$CVEC
 2344 014206  005737  002666                    TST     CLKFLD              ;L-CLOCK "TICKS"?
 2345 014212  001422                            BEQ     2$                  ;BRANCH IF NO "TICKS"
 2346 014214  013701  002642            1$:     MOV     PCLKCS,R1           ;GET POINTER TO CLOCK CONTROL STATUS REGISTER
 2347 014220  011137  002644                    MOV     (R1),PCSR           ;GET CLOCK CONTROL STATUS REGISTER
 2348 014224  016137  000004  002646            MOV     4(R1),VEC           ;GET CLOCK VECTOR ADDRESS
 2349 014232  016137  000006  002650            MOV     6(R1),HZ            ;GET CLOCK FREQUENCY
 2350 014240  022737  000074  002650            CMP     #60.,HZ             ;60 HZ.?
 2351 014246  001407                            BEQ     3$                  ;BRANCH - IF YES
 2352 014250  022737  000062  002650            CMP     #50.,HZ             ;50 HZ.?
 2353 014256  001420                            BEQ     4$                  ;BRANCH - IF YES
 2354 014260  005237  002652            2$:     INC     XITFLG              ;SET EXIT FLAG
 2355 014264  000475                            BR      8$                  ;EXIT
 2356 014266  005737  002420            3$:     TST     T.CNTLR             ;RL11?
 2357 014272  001404                            BEQ     9$                  ;BRANCH - IF NO
 2358 014274  012737  000030  002664            MOV     #24.,OPITIM         ;SET OPIMX FOR 60 HZ CLOCK & RL11
 2359 014302  000403                            BR      10$                 ;CONTINUE
 2360 014304  012737  000047  002664    9$:     MOV     #39.,OPITIM         ;SET OPIMX FOR 60 HZ CLOCK & RLV11
 2361 014312  005237  002656            10$:    INC     SIXTY               ;SET 60 HZ CLOCK INDICATOR
 2362 014316  000414                            BR      5$                  ;CHECK CLOCK TYPE
 2363 014320  005737  002420            4$:     TST     T.CNTLR             ;RL11?
 2364 014324  001404                            BEQ     11$                 ;BRANCH - IF NO
 2365 014326  012737  000024  002664            MOV     #20.,OPITIM         ;SET OPIMX FOR 50 HZ CLOCK & RL11
 2366 014334  000403                            BR      12$                 ;CONTINUE
 2367 014336  012737  000040  002664    11$:    MOV     #32.,OPITIM         ;SET OPIMX FOR 50 HZ CLOCK & RLV11
 2368 014344  005237  002654            12$:    INC     FIFTY               ;SET 50 HZ. CLOCK INDICATOR
 2369 014350  022737  000104  002646    5$:     CMP     #104,VEC            ;P-CLOCK?
 2370 014356  001016                            BNE     6$                  ;BRANCH - IF NO
```

GLOBAL SUBROUTINES

```
2371 014360  005237  002660                 INC     PCLOCK                  ;SET P-CLOCK INDICATOR
2372 014364                                 SETVEC  VEC,#CLKINT,#340        ;SET CLOCK INTERRUPT SERVICE ROUTINE
     014364  012746  000340                 MOV     #340,-(SP)
     014370  012746  014504                 MOV     #CLKINT,-(SP)
     014374  013746  002646                 MOV     VEC,-(SP)
     014400  012746  000003                 MOV     #3,-(SP)
     014404  104437                         TRAP    C$SVEC
     014406  062706  000010                 ADD     #10,SP
2373 014412  000422                         BR      8$                      ;EXIT
2374 014414  022737  000100  002646  6$:    CMP     #100,VEC                ;L-CLOCK?
2375 014422  001401                         BEQ     7$                      ;BRANCH - IF YES
2376 014424  000715                         BR      2$                      ;EXIT
2377 014426                         7$:     SETVEC  VEC,#CLKINT,#340        ;SET CLOCK INTERRUPT SERVICE ROUTINE
     014426  012746  000340                 MOV     #340,-(SP)
     014432  012746  014504                 MOV     #CLKINT,-(SP)
     014436  013746  002646                 MOV     VEC,-(SP)
     014442  012746  000003                 MOV     #3,-(SP)
     014446  104437                         TRAP    C$SVEC
     014450  062706  000010                 ADD     #10,SP
2378 014454  005037  002660                 CLR     PCLOCK                  ;INIT P-CLOCK INDICATOR
2379 014460  012601                 8$:     MOV     (SP)+,R1                ;RESTORE R1
2380 014462  000207                         RTS     PC                      ;RETURN
2381
2382 014464                         BGNSRV
2383 014464                         INTSRV:
2384
2385 014464  005237  002256                 INC     INTFLG                  ;SET INTERRUPT OCCURANCE FLAG
2386
2387 014470                         ENDSRV
     014470                         L10027:
     014470  000002                         RTI
2388
2389                                ;ROUTINE USED IN TIMING OPI
2390
2391 014472                         BGNSRV
2392 014472                         TIMSRV:
2393
2394 014472  005237  002256                 INC     INTFLG                  ;SET INTERRUPT INDICATOR FLAG
2395 014476  005077  166142                 CLR     @PCSR                   ;DISABLE CLOCK
2396
2397 014502                         ENDSRV
     014502                         L10030:
     014502  000002                         RTI
2398
2399 014504                         BGNSRV
2400 014504                         CLKINT:                                 ;CLOCK INTERRUPT SERVICE ROUTINE
2401
2402 014504  005337  002664                 DEC     OPITIM                  ;OPIMX EXPIRED?
2403 014510  001002                         BNE     1$                      ;BRANCH - IF NO
2404 014512  005077  166126                 CLR     @PCSR                   ;DISABLE CLOCK
2405 014516                         1$:
2406
2407 014516                         ENDSRV
     014516                         L10031:
     014516  000002                         RTI
2408
2409 014520                         BGNSRV
```

GLOBAL SUBROUTINES

```
2410 014520                            CLKTIK:                                 ;L-CLOCK "TICK" CHECK ROUTINE FOR LSI-11
2411
2412 014520  005237  002666                    INC     CLKFLD                  ;INCREMENT CLOCK FIELD TO INDICATE THAT
2413                                                                           ;/CLOCK IS "TICKING"
2414
2415 014524                            ENDSRV
     014524                            L10032:
     014524  000002                           RTI
2416
2417 014526                            CKERLT: INLOOP
     014526  104420                           TRAP    C$INLP
2418 014530                                    BCOMPLETE       99$
     014530  103427                           BCS     99$
2419 014532  005737  012434                    TST     DROP
2420 014536  001424                           BEQ     99$
2421 014540  005277  165660                    INC     @ERPOINT
2422 014544  027737  165654  012436           CMP     @ERPOINT,MERLMT
2423 014552  002416                           BLT     99$
2424 014554                                    PRINTF  #FRMT11
     014554  012746  012001                    MOV     #FRMT11,-(SP)
     014560  012746  000001                    MOV     #1,-(SP)
     014564  010600                           MOV     SP,R0
     014566  104417                           TRAP    C$PNTF
     014570  062706  000004                    ADD     #4,SP
2425 014574  004737  010522                    JSR     PC,LINE1
2426 014600                                    DODU    UNITST  ;DROP THIS UNIT
     014600  013700  002252                    MOV     UNITST,R0
     014604  104451                           TRAP    C$DODU
2427 014606                                    DOCLN
     014606  104444                           TRAP    C$DCLN
2428
2429 014610                            99$:
2430 014610  000205                           RTS     R5
2431
2432                                   .SBTTL  ROUTINE TO CHECK FOR CONTROLLER ERRORS
2433
2434                                   ;*************************************************************************
2435                                   ;*THIS ROUTINE WILL CHECK RLCS FOR ERRORS AND PRINT THEM
2436                                   ;*ACCORDINGLY.  IT WILL MERGE THE ERROR PRINTOUT WITH THE TEST
2437                                   ;*ERROR MESSAGE.
2438                                   ;*
2439                                   ;*ROUTINE USES R0,R1 AND PICKS HEADER FROM R3
2440                                   ;*
2441                                   ;*      CALL    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
2442                                   ;*
2443                                   ;*
2444                                   ;*
2445
2446 014612  005037  002236            CHERR:  CLR     T.CRC
2447 014616  032737  176000  002340            BIT     #176000,E.CS    ;ANY ERROR BITS SET?
2448 014624  001001                           BNE     2$              ;YES,FIND OUT WHICH
2449 014626  000205                           RTS     R5              ;NO EXIT
2450 014630  012701  004522            2$:     MOV     #EM100,R1       ;GET START OF STRING
2451 014634  005737  002340                    TST     E.CS            ;IS COMPOSITE ERROR SET?(BETTER BE)
2452 014640  100003                           BPL     99$             ;IT'S NOT SOMETHING IS WRONG
2453 014642  004537  015350                    JSR     R5,FIX          ;YES, PUT "COMP" IN STRING
2454 014646  003645                           COMP                    ;"COMP"
```

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```
2455 014650  032737  040000  002340  99$:    BIT     #DERR,E.CS      ;DRIVE ERROR SET?
2456 014656  001405                          BEQ     3$              ;NO, CONTINUE
2457 014660  005237  002422                  INC     DERFLG
2458 014664  004537  015350                  JSR     R5,FIX          ;YES, PUT "DRV" INTO STRING
2459 014670  003574                          DEMES                   ;"DRV"
2460 014672  032737  020000  002340  3$:     BIT     #NXM,E.CS       ;NON-EXISTENT MEMORY ERROR?
2461 014700  001403                          BEQ     4$              ;NO, CONTINUE
2462 014702  004537  015350                  JSR     R5,FIX          ;YES, PUT "NXM" INTO STRING
2463 014706  003601                          NXMMES                  ;"NXM"
2464 014710  032737  002000  002340  4$:     BIT     #OPI,E.CS       ;IS OPI SET?
2465 014716  001422                          BEQ     6$              ;NO, GO CHECK BITS 11 & 12
2466 014720  004537  015350                  JSR     R5,FIX          ;PUT "OPI" INTO STRING
2467 014724  003606                          OPIMES                  ;"OPI"
2468 014726  032737  004000  002340          BIT     #BIT11,E.CS     ;HEADERCRC ERROR?
2469 014734  001403                          BEQ     5$              ;NO, GO CHECK HEADER NOT FOUND
2470 014736  004537  015350                  JSR     R5,FIX          ;GO PUT "HCRC" IN STRING
2471 014742  003613                          HCRCMES                 ;"HCRC"
2472 014744  032737  010000  002340  5$:     BIT     #BIT12,E.CS     ;HEADER NOT FOUND?
2473 014752  001424                          BEQ     8$              ;NO, GO PUT "CRLF" IN STRING
2474 014754  004537  015350                  JSR     R5,FIX          ;PUT "HNF" IN STRING
2475 014760  003621                          HNFMES                  ;"HNF"
2476 014762  000420                          BR      8$              ;PUT "CRLF" IN STRING
2477 014764  032737  004000  002340  6$:     BIT     #BIT11,E.CS     ;DATA CRC ERROR?
2478 014772  001405                          BEQ     7$              ;NO, GO CHECK DATA LATE
2479 014774  005237  002236                  INC     T.CRC
2480 015000  004537  015350                  JSR     R5,FIX          ;PUT "DCK" IN STRING
2481 015004  003626                          DCKMES                  ;"DCK"
2482 015006  032737  010000  002340  7$:     BIT     #BIT12,E.CS     ;DATA LATE ERROR?
2483 015014  001403                          BEQ     8$              ;NO, GO PUT IN "CRLF"
2484 015016  004537  015350                  JSR     R5,FIX          ;PUT "DLT" IN STRING
2485 015022  003633                          DLTMES                  ;"DLT"
2486 015024  004537  015350          8$:     JSR     R5,FIX          ;PUT "CRLF" INTO STRING
2487 015030  003642                          MSCRLF                  ;"CRLF"
2488 015032  004537  015350                  JSR     R5,FIX          ;MOVE HEADER
2489 015036  000000                  RESTMS: .WORD   0               ;HEADER FROM TEST
2490 015040  105011                          CLRB    (R1)            ;PUT TERMINATOR IN
2491 015042                                  ERRDF   300.,LF,ERR6
     015042  104455                          TRAP    C$ERDF
     015044  000454                          .WORD   300
     015046  003640                          .WORD   LF
     015050  007760                          .WORD   ERR6
2492 015052  000205                          RTS     R5              ;EXIT ROUTINE
2493
2494                                 ;*****************************************************************
2495                                 ;* ROUTINE TO LOAD RLCS WITH FUNCTION TO BE PERFORMED
2496                                 ;*      CALL:   JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
2497                                 ;*              .WORD                   ;BITS TO BE LOADED, FUNCTION
2498                                 ;*                                      ;AND INTR ENABLE ONLY
2499                                 ;*
2500                                 ;
2501
2502 015054  032777  040000  165272  LDFUNC: BIT     #BIT14,@RLCS    ;DRIVE ERROR SET
2503 015062  001426                          BEQ     5$
2504 015064  017737  165270  002334          MOV     @RLDA,B.DA
2505 015072  012777  000013  165260          MOV     #13,@RLDA
2506 015100  012737  000200  002330          MOV     #200,B.CS
2507 015106  053737  002246  002330          BIS     DRIVE,B.CS
```

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```
2508 015114  013777  002330  165232          MOV     B.CS,@RLCS
2509 015122  032777  000200  165224  6$:     BIT     #200,@RLCS
2510 015130  001774                          BEQ     6$
2511 015132  013777  002334  165220          MOV     B.DA,@RLDA
2512 015140  012537  002260          5$:     MOV     (R5)+,LDCSR     ;GET BITS TO LOAD
2513 015144  010346                          MOV     R3,-(SP)        ;SAVE R3
2514 015146  042737  177661  002260          BIC     #177661,LDCSR   ;CLEAR ALL BUT FUNC & INTR EN
2515 015154  013737  002260  002372          MOV     LDCSR,FNDFNC    ;SAVE FUNCTION
2516 015162  042737  000100  002372          BIC     #INTEN,FNDFNC   ;ONLY FUNCTION
2517 015170  012703  015310                  MOV     #HDRLST,R3      ;GET HEADER LIST
2518 015174  006237  002372                  ASR     FNDFNC          ;ALIGN TO LEFT
2519 015200  001404                          BEQ     2$              ;IF EQUAL TO ZERO, SET R3
2520 015202  022323                  1$:     CMP     (R3)+,(R3)+     ;BUMP R3 BY 4
2521 015204  005337  002372                  DEC     FNDFNC          ;DEC FUNCTION
2522 015210  001374                          BNE     1$              ;FOUND IT? NO-GO BACK
2523 015212  032737  000100  002260  2$:     BIT     #INTEN,LDCSR    ;YES, DO WE WANT FLAG OR INTR?
2524 015220  001401                          BEQ     3$              ;FLAG BRANCH
2525 015222  005723                          TST     (R3)+           ;INTR POINT TO THAT ONE
2526 015224  011303                  3$:     MOV     (R3),R3         ;SET HEADER
2527 015226  010337  015036                  MOV     R3,RESTMS       ;SET UP HEADER
2528 015232  010337  002376                  MOV     R3,TRYFNC       ;SAVE HEADER FOR LATER
2529 015236  053737  002374  002260          BIS     XMEM,LDCSR      ;LOAD E.A. BITS
2530 015244  005037  002374                  CLR     XMEM            ;CLEAR OUT THE BITS
2531 015250  053737  002246  002260          BIS     DRIVE,LDCSR     ;SELECT DRIVE
2532 015256  052737  000200  002260          BIS     #200,LDCSR
2533 015264  013777  002260  165062          MOV     LDCSR,@RLCS     ;LOAD FUNCTION
2534 015272  004537  015362                  JSR     R5,BEFORE       ;READ REGISTERS
2535 015276  042777  000200  165050  4$:     BIC     #200,@RLCS      ;ISSUE COMMAND
2536 015304  012603                          MOV     (SP)+,R3        ;RESTORE R3
2537 015306  000205                          RTS     R5              ;EXIT
2538
2539 015310  003705                  HDRLST: NOPMES
2540 015312  003733                          NOPINT
2541 015314  003762                          WCKMES
2542 015316  004016                          WCKINT
2543 015320  004171                          GSTMES
2544 015322  004221                          GSTINT
2545 015324  004130                          SEKMES
2546 015326  004150                          SEKINT
2547 015330  004053                          RHDMES
2548 015332  004103                          RHDINT
2549 015334  004311                          WRTMES
2550 015336  004332                          WRTINT
2551 015340  004251                          RDDMES
2552 015342  004271                          RDDINT
2553 015344  004353                          RDNMES
2554 015346  004401                          RDNINT
2555
2556                                ;*************************************************************************
2557                                ;*ROUTINE TO MOVE ASCII STRINGS
2558                                ;*USES REGISTERS R1 - WHERE STRING IS BEING BUILT
2559                                ;*
2560                                ;*      CALL    JSR     R5,FIX
2561                                ;*              .WORD           ;ADDRESS OF STRING TO MOVE
2562
2563 015350  012504                  FIX:    MOV     (R5)+,R4        ;GET ADDRESS AND MOVE RETURN
2564 015352  112421                  1$:     MOVB    (R4)+,(R1)+     ;GET BYTE AND UPDATE
```

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```
2565 015354  001376                          BNE     1$              ;WATCH 0 BYTE TERMINATOR
2566 015356  105741                          TSTB    -(R1)           ;BACK UP OVER ZERO BYTE
2567 015360  000205                          RTS     R5              ;EXIT
2568
2569                                 ;ROUTINE TO READ REGISTERS PRIOR TO OPERATION
2570                                 ;CALL:  JSR R5,BEFORE
2571
2572 015362  017737  164766  002330 BEFORE: MOV     @RLCS,B.CS      ;READ CS
2573 015370  017737  164762  002332         MOV     @RLBA,B.BA      ;      BA
2574 015376  017737  164756  002334         MOV     @RLDA,B.DA      ;      DA
2575 015404  017737  164752  002336         MOV     @RLMP,B.MP      ;      MP
2576 015412  000205                          RTS     R5
2577
2578                                 ;ROUTINE TO READ REGISTERS AT TIME OF ERROR
2579                                 ;CALL:  JSR R5,AFTER
2580
2581 015414  017737  164734  002340 AFTER:  MOV     @RLCS,E.CS      ;READ CS
2582 015422  017737  164730  002342         MOV     @RLBA,E.BA      ;      BA
2583 015430  017737  164724  002344         MOV     @RLDA,E.DA      ;      DA
2584 015436  017737  164720  002346         MOV     @RLMP,E.MP      ;      MP
2585 015444  017737  164712  002350         MOV     @RLMP,E.MP1     ;      MP
2586 015452  017737  164704  002352         MOV     @RLMP,E.MP2     ;      MP
2587 015460  000205                          RTS     R5
2588
2589 015462  010046                 SIMBCC: MOV     R0,-(SP)        ;SAVE R0
2590 015464  010146                         MOV     R1,-(SP)        ;SAVE R1
2591 015466  010246                         MOV     R2,-(SP)        ;SAVE R2
2592 015470  012537  002304                 MOV     (R5)+,TEMP2     ;GET NUMBER OF BITS
2593 015474  012537  002306                 MOV     (R5)+,TEMP3     ;GET DATA FOR CRC CALCULATION
2594 015500  012537  002310                 MOV     (R5)+,TEMP4     ;GET STARTING CRC
2595 015504  005037  002266          1$:    CLR     BCCFBK          ;
2596 015510  013700  002310                 MOV     TEMP4,R0        ;GET PRESENT CRC
2597 015514  006037  002306                 ROR     TEMP3           ;ROTATE NEW DATA
2598 015520  005500                         ADC     R0              ;MERGE NEW WITH OLD
2599 015522  032700  000001                 BIT     #1,R0           ;BIT 0 SET
2600 015526  001402                         BEQ     2$              ;IF NOT CONTINUE
2601 015530  005137  002266                 COM     BCCFBK          ;
2602 015534  013700  002264          2$:    MOV     XPOLY,R0        ;GET CRC POLYNOMIAL (CRC-16)
2603 015540  005100                         COM     R0              ;COMPLEMENT POLYNOMIAL
2604 015542  040037  002266                 BIC     R0,BCCFBK
2605 015546  000241                         CLC                     ;CLEAR CARRY
2606 015550  006037  002310                 ROR     TEMP4
2607 015554  013700  002266                 MOV     BCCFBK,R0
2608 015560  013701  002310                 MOV     TEMP4,R1
2609 015564  010102                         MOV     R1,R2
2610 015566  040100                         BIC     R1,R0
2611 015570  043702  002266                 BIC     BCCFBK,R2
2612 015574  050200                         BIS     R2,R0
2613 015576  043737  002264  002310         BIC     XPOLY,TEMP4
2614 015604  050037  002310                 BIS     R0,TEMP4
2615 015610  005337  002304                 DEC     TEMP2
2616 015614  001333                         BNE     1$
2617
2618 015616  013737  002310  002270         MOV     TEMP4,CALBCC
2619 015624  012602                         MOV     (SP)+,R2
2620 015626  012601                         MOV     (SP)+,R1
2621 015630  012600                         MOV     (SP)+,R0
```

```
ROUTINE TO CHECK FOR CONTROLLER ERRORS

 2622 015632  000205                          RTS     R5              ;RETURN
 2623
 2624                                ;ROUTINE TO WAIT FOR DRIVE READY
 2625
 2626 015634  012701  000144          WTDRDY: MOV     #100.,R1
 2627 015640  032777  000001  164506  1$:     BIT     #DRDY,@RLCS
 2628 015646  001013                          BNE     2$
 2629
 2630 015650                                   WAITUS  #20.
 2631 015662  005301                          DEC     R1
 2632 015664  001365                          BNE     1$
 2633
 2634 015666                                   ERRDF   200.,DRTIM,ERR5
      015666  104455                          TRAP    C$ERDF
      015670  000310                          .WORD   200
      015672  003546                          .WORD   DRTIM
      015674  007722                          .WORD   ERR5
 2635
 2636 015676  000205                2$:     RTS     R5
 2637
 2638                                ;ROUTINE TO WAIT FOR CONTROLLER
 2639
 2640 015700  012701  000620          WTCRDY: MOV     #400.,R1
 2641 015704  032777  000200  164442  1$:     BIT     #CRDY,@RLCS
 2642 015712  001016                          BNE     2$
 2643
 2644 015714                                   WAITUS  #20.
 2645 015726  005301                          DEC     R1
 2646 015730  001365                          BNE     1$
 2647 015732  004537  015414                JSR     R5,AFTER
 2648
 2649 015736                                   ERRDF   100.,CRTIM,ERR5
      015736  104455                          TRAP    C$ERDF
      015740  000144                          .WORD   100
      015742  003521                          .WORD   CRTIM
      015744  007722                          .WORD   ERR5
 2650 015746  000205                          RTS     R5
 2651
 2652 015750  004537  015414          2$:     JSR     R5,AFTER
 2653 015754  000205                          RTS     R5
 2654
 2655 015756  005237  002254          TRPHAN: INC     TRPFLG
 2656 015762  000002                          RTI
 2657
 2658 015764                          HDHOME:
 2659
 2660 015764                                   BGNSEG                          ;**START OF SEGMENT**
      015764  104404                          TRAP    C$BSEG
 2661                                ;ISSUE DRIVE RESET
 2662
 2663 015766  012737  000001  002400          MOV     #1,ERFLG        ;SET ERROR FLAG
 2664 015774  012777  000013  164356          MOV     #DRST!MK!GSBIT,@RLDA
 2665 016002  004537  015054                JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 2666 016006  000004                          GSTAT
 2667 016010  004537  015700                JSR     R5,WTCRDY
 2668 016014                                   ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      016014  104410                          TRAP    C$ESCAPE
```

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```
                016016  000216                  .WORD   10000$-.
2669            016020  004537  014612          JSR     R5,CHERR        ;CHECK CNTLR FOR ERRORS
2670            016024                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                016024  104410                  TRAP    C$ESCAPE
                016026  000206                  .WORD   10000$-.
2671
2672            016030  004537  015054          JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
2673            016034  000010                  RDHDR
2674            016036                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                016036  104410                  TRAP    C$ESCAPE
                016040  000174                  .WORD   10000$-.
2675            016042  004537  015700          JSR     R5,WTCRDY
2676            016046                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                016046  104410                  TRAP    C$ESCAPE
                016050  000164                  .WORD   10000$-.
2677
2678            016052  004537  014612          JSR     R5,CHERR        ;CHECK CNTLR FOR ERRORS
2679            016056                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                016056  104410                  TRAP    C$ESCAPE
                016060  000154                  .WORD   10000$-.
2680
2681            016062  013737  002346  002272  MOV     E.MP,TMPO       ;GET HEADER
2682            016070  042737  000077  002272  BIC     #77,TMPO
2683            016076  001424                  BEQ     99$             ;SEEK IS NOT NECESSARY
2684            016100  042737  000100  002272  BIC     #100,TMPO
2685            016106  012777  000001  164244  MOV     #1K,@RLDA       ;SET TO SEEK
2686            016114  053777  002272  164236  BIS     TMPO,@RLDA      ;SET IN DIFFERENCE
2687
2688            016122  004537  015054          JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
2689            016126  000006                  SEEK
2690            016130  004537  015700          JSR     R5,WTCRDY
2691            016134                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                016134  104410                  TRAP    C$ESCAPE
                016136  000076                  .WORD   10000$-.
2692
2693            016140  004537  014612          JSR     R5,CHERR        ;CHECK CNTLR FOR ERRORS
2694            016144                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                016144  104410                  TRAP    C$ESCAPE
                016146  000066                  .WORD   10000$-.
2695
2696            016150  004537  015054  99$:    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
2697            016154  000010                  RDHDR
2698            016156  004537  015700          JSR     R5,WTCRDY
2699            016162                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                016162  104410                  TRAP    C$ESCAPE
                016164  000050                  .WORD   10000$-.
2700            016166  004537  014612          JSR     R5,CHERR
2701            016172                          ESCAPE  SEG
                016172  104410                  TRAP    C$ESCAPE
                016174  000040                  .WORD   10000$-.
2702
2703            016176  013737  002346  002272  MOV     E.MP,TMPO       ;GET HEADER
2704            016204  043737  002262  002272  BIC     SECMSK,TMPO     ;IGNORE SECTOR
2705            016212  001404                  BEQ     1$              ;ON ZERO
2706
2707            016214                          ERRDF   400.,SKHOME,ERRO        ;CAN'T SEEK TO TRACK 0
                016214  104455                  TRAP    C$ERDF
```

```
ROUTINE TO CHECK FOR CONTROLLER ERRORS

        016216  000620                          .WORD   400
        016220  004430                          .WORD   SKHOME
        016222  007510                          .WORD   ERRO
   2708
   2709 016224                          1$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
        016224  104410                          TRAP    C$ESCAPE
        016226  000006                          .WORD   10000$-.
   2710
   2711 016230  005037  002400                 CLR     ERFLG           ;INDICATE SUCCESS BACK TO MAIN PROGRAM
   2712
   2713 016234                                  ENDSEG                          ;**END OF SEGMENT**
        016234                          10000$:
        016234  104405                          TRAP    C$ESEG
   2714
   2715 016236  000207                          RTS     PC
   2716
   2717 016240                                  ENDMOD
   2718
   2719                                 .SBTTL  **TEST 1** - WRITE FUNCTION
   2720
   2721 016240                          BGNTST                                  ;**START OF TEST**
   2722
   2723 016240                          STARS
                                        ;;**********************************************************************
   2724                                 ;CHECK OF WRITE LOGIC UNDER FLAG MODE, WE WILL FIRST ISSUE A
   2725                                 ;READ HEADER SO THAT WE DON'T WRITE ON THE BAD SECTOR
   2726                                 ;FILE TRACK.  WE WILL WRITE A FULL SECTOR (128 WORDS) FROM
   2727                                 ;MEMORY (BUF).  WE CHECK THAT NO ERRORS OCCUR.  IF WE
   2728                                 ;HAVE A DRIVE ERROR WE WILL DO A "GET STATUS" TO SEE
   2729                                 ;IF WRITE PROTECT IS SET IF IT IS WE WILL ABORT THE
   2730                                 ;TEST.  AN ERROR ON THE WRITE WILL LOOP ON JUST THE
   2731                                 ;WRITE PORTION.  LOOP ON TEST WILL READ HEADER, SEEK (IF
   2732                                 ;NECESSARY) AND WRITE.
   2733 016240                          STARS
                                        ;;**********************************************************************
   2734
   2735 016240  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
   2736 016244                                  CKERFG                  ;HEADS GO HOME OKAY
        016252  104432                          TRAP    C$EXIT
        016254  000126                          .WORD   L10033-.
   2737
   2738 016256                                  BGNSEG                          ;**START OF SEGMENT**
        016256  104404                          TRAP    C$BSEG
   2739
   2740 016260                          3$:
   2741 016260  005077  164074                  CLR     @RLDA           ;SET DISK ADDRESS
   2742 016264  012777  177600  164070          MOV     #-128.,@RLMP    ;WORD COUNT
   2743 016272  012777  003426  164056          MOV     #BUF,@RLBA      ;BUS ADDRESS
   2744 016300  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
   2745 016304  000012                          WRITE                   ;WRITE
   2746
   2747 016306  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
   2748 016312                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
        016312  104410                          TRAP    C$ESCAPE
        016314  000064                          .WORD   10001$-.
   2749
   2750 016316  032777  040000  164030          BIT     #DERR,@RLCS     ;DRIVE ERROR SET?
```

```
**TEST 1** - WRITE FUNCTION

2751 016324  001425                          BEQ     4$                ;BRANCH IF NOT
2752
2753 016326  012777  000003  164024          MOV     #MK!GSBIT,@RLDA ;SET GET STATUS OF DRIVE
2754 016334  004537  015054                  JSR     R5,LDFUNC                 ;LOAD THE FUNCTION IN NEXT WORD
2755 016340  000004                          GSTAT                     ;GET STATUS
2756 016342  004537  015700                  JSR     R5,WTCRDY         ;WAIT FOR CONTROLLER READY
2757 016346                                  ESCAPE  SEG               ;CHECK FOR FL:LOE, ELSE EXIT SEG
     016346  104410                          TRAP    C$ESCAPE
     016350  000030                          .WORD   10001$-.
2758
2759 016352  013737  002346  002300          MOV     E.MP,GDDAT        ;READ DRIVE STATUS
2760 016360  032737  020000  002300          BIT     #BIT13,GDDAT      ;WRITE LOCK ERROR?
2761 016366  001404                          BEQ     4$                ;NO, BRANCH
2762
2763 016370                                  ERRSF   3.,WRLOCK,ERRO            ;WRITE LOCK ERROR
     016370  104454                          TRAP    C$ERSF
     016372  000003                          .WORD   3
     016374  004452                          .WORD   WRLOCK
     016376  007510                          .WORD   ERRO
2764 016400                          4$:
2765
2766 016400                                  ENDSEG                            ;**END OF SEGMENT**
     016400                          10001$:
     016400  104405                          TRAP    C$ESEG
2767 016402                          ENDTST                            ;**END OF TEST**
     016402                          L10033:
     016402  104401                          TRAP    C$ETST
2768
2769                                 .SBTTL  **TEST 2** - WRITE FUNCTION INTERRUPT
2770
2771 016404                          BGNTST                            ;**START OF TEST**
2772
2773 016404                          STARS
                                     ;;*****************************************************************
2774                                 ;CHECK OF WRITE LOGIC UNDER INTERRUPT MODE, WE WILL ISSUE A
2775                                 ;READ HEADER SO THAT WE DON'T WRITE ON THE BAD SECTOR FILE
2776                                 ;TRACK.  WE WILL WRITE A FULL SECTOR (128 WORDS) FROM MEMORY (BUF).
2777                                 ;WE CHECK THAT NO ERRORS OCCUR.  WE DO NOT CHECK RLDA OR RLBA
2778                                 ;INCREMENT AT THIS TIME.
2779 016404                          STARS
                                     ;;*****************************************************************
2780
2781 016404  004737  015764                  JSR     PC,HDHOME         ;HEADS OVER TRACK 0
2782 016410                                  CKERFG                    ;HEADS GO HOME OKAY
     016416  104432                          TRAP    C$EXIT
     016420  000112                          .WORD   L10034-.
2783
2784 016422                                  BGNSEG                            ;**START OF SEGMENT**
     016422  104404                          TRAP    C$BSEG
2785
2786 016424  005037  002256                  CLR     INTFLG            ;CLEAR INTERRUPT OCCURANCE FLAG
2787 016430  005077  163724                  CLR     @RLDA
2788 016434  012777  177600  163720          MOV     #-128.,@RLMP      ;SET UP WORD COUNT
2789 016442  012777  003426  163706          MOV     #BUF,@RLBA        ;SET UP BUS ADDRESS
2790
2791 016450                                  SETPRI  #PRI00            ;PRIORITY TO 0
     016450  012700  000000                  MOV     #PRI00,R0
```

```
**TEST 2** - WRITE FUNCTION INTERRUPT

     016454  104441                         TRAP    C$SPRI
2792 016456  004537  015054                 JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
2793 016462  000112                         WRITE!INTEN                     ;WRITE UNDER INTERRUPT
2794 016464  004537  015700                 JSR     R5,WTCRDY               ;WAIT FOR INTERRUPT
2795 016470                                 ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     016470  104410                         TRAP    C$ESCAPE
     016472  000036                         .WORD   10000$-.
2796
2797                                 ;      SETPRI  #PRI07                  ;SET PRIORITY TO 7              ;JSD REV A
2798 016474                                 SETPRI  #PRI06                  ;SET PRIORITY TO 6              ;JSD REV A
     016474  012700  000300                 MOV     #PRI06,R0
     016500  104441                         TRAP    C$SPRI
2799 016502  005737  002256                 TST     INTFLG                  ;DID INTERRUPT OCCUR?
2800 016506  001004                         BNE     2$                      ;YES-BRANCH NO-REPORT
2801
2802 016510                                 ERRDF   4.,EM17,ERR0            ;WRITE DID NOT INTERRUPT
     016510  104455                         TRAP    C$ERDF
     016512  000004                         .WORD   4
     016514  005322                         .WORD   EM17
     016516  007510                         .WORD   ERR0
2803 016520                         2$:     ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     016520  104410                         TRAP    C$ESCAPE
     016522  000006                         .WORD   10000$-.
2804
2805 016524  004537  014612                 JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
2806
2807 016530                                 ENDSEG                          ;**END OF SEGMENT**
     016530                         10000$:
     016530  104405                         TRAP    C$ESEG
2808 016532                         ENDTST                                  ;**END OF TEST**
     016532                         L10034:
     016532  104401                         TRAP    C$ETST
2809
2810                                 .SBTTL  **TEST 3** - PROPER INCREMENT OF RLBA ON WRITE
2811
2812 016534                         BGNTST                                  ;**START OF TEST**
2813
2814 016534                         STARS
                                    ;;*************************************************************************
2815                                 ;CHECK THAT THE RLBA WILL INCREMENT PROPERLY AFTER THE
2816                                 ;WRITE WAS FINISHED THE RLBA SHOULD BE 128 WORDS (256 BYTES)
2817                                 ;CREATER.  STARTING RLBA IS "BUF", ENDING SHOULD BE "BUF + 256."
2818                                 ;WE WILL MONITOR ALL ERRORS AND REPORT THEM ACCORDINGLY
2819 016534                         STARS
                                    ;;*************************************************************************
2820
2821 016534  004737  015764                 JSR     PC,HDHOME               ;HEADS OVER TRACK 0
2822 016540                                 CKERFG                          ;HEADS GO HOME OKAY
     016546  104432                         TRAP    C$EXIT
     016550  000116                         .WORD   L10035-.
2823
2824 016552                                 BGNSEG                          ;**START OF SEGMENT**
     016552  104404                         TRAP    C$BSEG
2825
2826 016554                         3$:
2827 016554  005077  163600                 CLR     @RLDA
2828 016560  012777  003426  163570         MOV     #BUF,@RLBA              ;SET UP BUS ADDRESS
```

```
**TEST 3** - PROPER INCREMENT OF RLBA ON WRITE

2829 016566  012777  177600  163566          MOV     #-128.,@RLMP    ;WORD COUNT
2830 016574  012737  003426  002300          MOV     #BUF,GDDAT      ;FORM EXPECTED BUS ADDRESS
2831 016602  062737  000400  002300          ADD     #256.,GDDAT     ;AFTER WRITE
2832
2833 016610  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
2834 016614  000012                          WRITE                   ;WRITE
2835 016616  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
2836 016622                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     016622  104410                          TRAP    C$ESCAPE
     016624  000040                          .WORD   10000$-.
2837
2838 016626  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
2839 016632                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     016632  104410                          TRAP    C$ESCAPE
     016634  000030                          .WORD   10000$-.
2840 016636  017737  163514  002302          MOV     @RLBA,BDDAT     ;READ 'RLBA' FOR PRESENT ADDRESS
2841 016644  023737  002302  002300          CMP     BDDAT,GDDAT     ;DID 'BA' INCREMENT PROPERLY?
2842 016652  001404                          BEQ     2$              ;YES, CONTINUE
2843
2844 016654                                  ERRDF   5.,EM20,ERR4    ;BA DID NOT INCREMENT
     016654  104455                          TRAP    C$ERDF
     016656  000005                          .WORD   5
     016660  005346                          .WORD   EM20
     016662  007654                          .WORD   ERR4
2845
2846 016664                          2$:
2847
2848 016664                                  ENDSEG                          ;**END OF SEGMENT**
     016664                          10000$:
     016664  104405                          TRAP    C$ESEG
2849 016666                          ENDTST                          ;**END OF TEST**
     016666                          L10035:
     016666  104401                          TRAP    C$ETST
2850
2851                                 .SBTTL  **TEST 4** - PROPER INCREMENT OF RLDA ON WRITE
2852
2853 016670                          BGNTST                                  ;**START OF TEST**
2854
2855 016670                          STARS
                                     ;;*************************************************************************
2856                                 ;CHECK THAT THE SECTOR INCREMENTS AFTER THE WRITE WAS FINISHED.
2857                                 ;WE RANDOMLY PICK A SECTOR (OTHER THAN LAST TRACK) AND ISSUE
2858                                 ;A FULL SECTOR WRITE THE RLDA SHOULD REFLECT AN INCREMENT
2859                                 ;OF THE SECOTR.  "GDDAT" WAS THE EXPECTED RLDA.
2860 016670                          STARS
                                     ;;*************************************************************************
2861
2862 016670  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
2863 016674                                  CKERFG                  ;HEADS GO HOME OKAY
     016702  104432                          TRAP    C$EXIT
     016704  000114                          .WORD   L10036-.
2864
2865 016706                                  BGNSEG                          ;**START OF SEGMENT**
     016706  104404                          TRAP    C$BSEG
2866
2867 016710                          3$:
2868 016710  005037  002300                  CLR     GDDAT
```

```
**TEST 4** - PROPER INCREMENT OF RLDA ON WRITE

2869 016714  013777  002300  163436          MOV     GDDAT,@RLDA     ;SETUP DISK ADDRESS
2870 016722  005237  002300                  INC     GDDAT           ;CREATE EXPECTED SECTOR
2871 016726  012777  177600  163426          MOV     #-128.,@RLMP    ;WORD COUNT
2872 016734  012777  003426  163414          MOV     #BUF,@RLBA      ;SETUP BUS ADDRESS
2873
2874 016742  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
2875 016746  000012                          WRITE                   ;WRITE
2876 016750  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
2877 016754                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     016754  104410                          TRAP    C$ESCAPE
     016756  000040                          .WORD   10000$-.
2878
2879 016760  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
2880 016764                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     016764  104410                          TRAP    C$ESCAPE
     016766  000030                          .WORD   10000$-.
2881
2882 016770  013737  002344  002302          MOV     E.DA,BDDAT      ;READ DISK ADDRESS
2883 016776  023737  002300  002302          CMP     GDDAT,BDDAT     ;DID SECTOR INCREMENT PROPERLY
2884 017004  001404                          BEQ     2$              ;YES, BRANCH  NO, REPORT ERROR
2885
2886 017006                                  ERRDF   6.,EM21,ERR4    ;DA DID NOT INCREMENT
     017006  104455                          TRAP    C$ERDF
     017010  000006                          .WORD   6
     017012  005414                          .WORD   EM21
     017014  007654                          .WORD   ERR4
2887
2888 017016                          2$:
2889
2890 017016                                  ENDSEG                          ;**END OF SEGMENT**
     017016                          10000$:
     017016  104405                          TRAP    C$ESEG
2891 017020                          ENDTST                                  ;**END OF TEST**
     017020                          L10036:
     017020  104401                          TRAP    C$ETST
2892
2893                                 .SBTTL  **TEST 5** - FORCE HEADER NOT FOUND WITH WRITE
2894
2895 017022                          BGNTST                                  ;**START OF TEST**
2896
2897 017022                          STARS
                                     ;;*************************************************************************
2898                                 ;FORCE HEADER NOT FOUND ERROR TO OCCUR.  THIS IS DONE
2899                                 ;BY SETTING SECTOR 40 OF THE RLDA AND ISSUING A
2900                                 ;WRITE.  SECTOR 40 DOES NOT EXIST ON THE RL01 PACK
2901                                 ;THEREFORE HDR NT FOUND SHOULD SET.
2902 017022                          STARS
                                     ;;*************************************************************************
2903
2904 017022  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
2905 017026                                  CKERFG                  ;HEADS GO HOME OKAY
     017034  104432                          TRAP    C$EXIT
     017036  000120                          .WORD   L10037-.
2906
2907 017040                                  BGNSEG                          ;**START OF SEGMENT**
     017040  104404                          TRAP    C$BSEG
2908
```

```
**TEST 5** - FORCE HEADER NOT FOUND WITH WRITE

 2909 017042  012777  000050  163310          MOV     #40.,@RLDA      ;INSURE NOT TO FIND HEADER BY
 2910 017050  012777  003426  163300          MOV     #BUF,@RLBA      ;SETTING SECTOR 40 OF CYL. ADDR.
 2911 017056  012777  177777  163276          MOV     #-1,@RLMP       ;WORD COUNT
 2912
 2913 017064  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 2914 017070  000012                          WRITE                   ;WRITE
 2915 017072  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
 2916 017076                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      017076  104410                          TRAP    C$ESCAPE
      017100  000054                          .WORD   10000$-.
 2917
 2918 017102  013737  002340  002272          MOV     E.CS,TMP0       ;GET RLCS
 2919 017110  042737  001777  002272          BIC     #1777,TMP0      ;SAVE ERROR BITS
 2920 017116  022737  112000  002272          CMP     #BIT15!BIT12!BIT10,TMP0  ;HDR NOT FOUND SET.
 2921 017124  001402                          BEQ     1$              ;YES, CONTINUE
 2922
 2923 017126  004537  014612                  JSR     R5,CHERR
 2924 017132                          1$:     CKLOOP
      017132  104406                          TRAP    C$CLP1
 2925
 2926 017134  022737  112000  002272          CMP     #BIT15!BIT12!BIT10,TMP0
 2927 017142  001404                          BEQ     2$
 2928 017144                                  ERRDF   23.,EM10,ERR0
      017144  104455                          TRAP    C$ERDF
      017146  000027                          .WORD   23
      017150  005064                          .WORD   EM10
      017152  007510                          .WORD   ERR0
 2929                                                                 ;WHEN FORCED
 2930 017154                          2$:
 2931
 2932 017154                                  ENDSEG                          ;**END OF SEGMENT**
      017154                          10000$:
      017154  104405                          TRAP    C$ESEG
 2933 017156                          ENDTST                          ;**END OF TEST**
      017156                          L10037:
      017156  104401                          TRAP    C$ETST
 2934
 2935                                 .SBTTL  **TEST 6** - FORCE HEADER NOT FOUND WITH WRITE INTERRUPT
 2936
 2937 017160                          BGNTST                          ;**START OF TEST**
 2938
 2939 017160                          STARS
                                      ;;**************************************  **********************************
 2940                                 ;TEST THAT HEADER NOT FOUND ERROR WILL GENERATE AN INTERRUPT
 2941                                 ;ON OCCURRENCE.  HEADER NOT FOUND WILL BE FORCED BY SETTING
 2942                                 ;SECTOR 40 OF RLDA AND ISSUEING A WRITE
 2943 017160                          STARS
                                      ;;**************************************************************************

 2944
 2945 017160  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 2946 017164                                  CKERFG                  ;HEADS GO HOME OKAY
      017172  104432                          TRAP    C$EXIT
      017174  000160                          .WORD   L10040-.
 2947
 2948 017176                                  BGNSEG                          ;**START OF SEGMENT**
      017176  104404                          TRAP    C$BSEG
 2949
```

TEST 6 - FORCE HEADER NOT FOUND WITH WRITE INTERRUPT

```
2950 017200                                    SETPRI  #PRI00
     017200  012700  000000                    MOV     #PRI00,R0
     017204  104441                            TRAP    C$SPRI
2951 017206  005037  002256                    CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE FLAG
2952 017212  012777  000050  163140            MOV     #40.,@RLDA      ;INSURE NOT TO FIND HEADER BY
2953 017220  012777  003426  163130            MOV     #BUF,@RLBA      ;SETTING SECTOR 40 OF CYL. ADDR.
2954 017226  012777  177777  163126            MOV     #-1,@RLMP       ;WORD COUNT
2955
2956 017234  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
2957 017240  000112                            WRITE!INTEN             ;WRITE
2958 017242  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
2959 017246                                    CKLOOP
     017246  104406                            TRAP    C$CLP1
2960                                 ;         SETPRI  #PRI07                                  ;JSD REV A
2961 017250                                    SETPRI  #PRI06                                  ;JSD REV A
     017250  012700  000300                    MOV     #PRI06,R0
     017254  104441                            TRAP    C$SPRI
2962
2963 017256  005737  002256                    TST     INTFLG          ;DID INTERRUPT OCCUR
2964 017262  001004                            BNE     2$              ;YES OKAY
2965
2966 017264                                    ERRDF   24.,EM43,ERR0   ;NO INTERRUPT FROM OPI
     017264  104455                            TRAP    C$ERDF
     017266  000030                            .WORD   24
     017270  006461                            .WORD   EM43
     017272  007510                            .WORD   ERR0
2967
2968 017274                           2$:      ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     017274  104410                            TRAP    C$ESCAPE
     017276  000054                            .WORD   10000$-.
2969
2970 017300  013737  002340  002272            MOV     E.CS,TMP0       ;GET RLCS
2971 017306  042737  001777  002272            BIC     #1777,TMP0      ;SAVE ERROR BITS
2972 017314  022737  112000  002272            CMP     #BIT15!BIT12!BIT10,TMP0  ;WDR NOT FOUND SET.
2973 017322  001402                            BEQ     1$              ;YES, CONTINUE
2974
2975 017324  004537  014612                    JSR     R5,CHERR
2976 017330                           1$:      CKLOOP
     017330  104406                            TRAP    C$CLP1
2977
2978 017332  022737  112000  002272            CMP     #BIT15!BIT12!BIT10,TMP0
2979 017340  001404                            BEQ     3$
2980 017342                                    ERRDF   25.,EM10,ERR0
     017342  104455                            TRAP    C$ERDF
     017344  000031                            .WORD   25
     017346  005064                            .WORD   EM10
     017350  007510                            .WORD   ERR0
2981                                                                ;WHEN FORCED
2982 017352                           3$:
2983
2984 017352                                    ENDSEG                          ;**END OF SEGMENT**
     017352                           10000$:
     017352  104405                            TRAP    C$ESEG
2985 017354                           ENDTST                           ;**END OF TEST**
     017354                           L10040:
     017354  104401                            TRAP    C$ETST
2986
```

TEST 7 - CHECK OPI TIME WITH HDR NT FND

```
                                .SBTTL  **TEST 7** - CHECK OPI TIME WITH HDR NT FND

017356                          BGNTST                                  ;**START OF TEST**

017356                          STARS
                                ;;*************************************************************************
                                ;CHECK OPI TIME IT SHOULD BE AROUND 200 MILLISECONDS (ON UNIBUS)
                                ;CHECK THIS BY TIMING OPI ON A FORCED HEADER NOT FOUND
                                ;ISSUE WRITE WITH SECTOR 40 SET IN THE DISK ADDRESS
017356                          STARS
                                ;;*************************************************************************

017356  004737  014110                  JSR     PC,SETCLK       ;CALL INITIALIZE CLOCK ROUTINE
017362  005737  002652                  TST     XITFLG          ;EXIT?
017366  001402                          BEQ     1$              ;BRANCH - IF NO
                                ;       PRINTB  #FRMT18         ;ELSE, PRINT MSG. "TEST 7 CANNOT BE PERFORMED...        ;JSD
REV A
                                                                ;/CLOCK IS NOT AVAILABLE"
017370  000137  017742                  JMP     8$              ;EXIT
017374  004737  015764          1$:     JSR     PC,HDHOME       ;HEADS OVER TRACK 0
017400                                  CKERFG                  ;HEADS GO HOME OKAY
017406  104432                          TRAP    C$EXIT
017410  000346                          .WORD   L10041-.

017412                                  BGNSEG                  ;##START OF SEGMENT##
017412  104404                          TRAP    C$BSEG

017414                                  CLRVEC  BVEC            ;CLEAR PRESENT INTERRUPT VECTOR
017414  013700  002366                  MOV     BVEC,R0
017420  104436                          TRAP    C$CVEC
017422                                  SETVEC  BVEC,#TIMSRV,#340       ;SET INTR. VEC. WITH DISABLE CLOCK
017422  012746  000340                  MOV     #340,-(SP)
017426  012746  014472                  MOV     #TIMSRV,-(SP)
017432  013746  002366                  MOV     BVEC,-(SP)
017436  012746  000003                  MOV     #3,-(SP)
017442  104437                          TRAP    C$SVEC
017444  062706  000010                  ADD     #10,SP
017450                                  SETPRI  #PRIOO
017450  012700  000000                  MOV     #PRIOO,R0
017454  104441                          TRAP    C$SPRI
017456  005037  002256                  CLR     INTFLG          ;CLEAR INTERRUPT FLAG
017462  012777  000050  162670          MOV     #40.,@RLDA      ;SET UP FOR HDR NT FND
017470  012777  003426  162660          MOV     #BUF,@RLBA      ;BUS ADDRESS
017476  012777  177777  162656          MOV     #-1,@RLMP       ;WORD COUNT
017504  013737  002664  002302          MOV     OPITIM,BDDAT    ;GET OPIMX FOR WORST CASE
017512  013701  002644                  MOV     PCSR,R1         ;GET CSR
017516  005737  002660                  TST     PCLOCK          ;USING THE P-CLOCK?
017522  001404                          BEQ     6$              ;BRANCH - IF NO
017524  012711  000014                  MOV     #14,(R1)        ;SET P-CLOCK,  REPEAT-INT,LINE FREQ.
017530  005061  000002                  CLR     2(R1)           ;INIT COUNT BUFFER REGISTER
017534  004537  015054          6$:     JSR     R5,LDFUNC       ;LOAD THE FUNCTION IN THE NEXT WORD
017540  000112                          WRITE!INTEN             ;WRITE UNDER INTERRUPT
017542  013700  002664                  MOV     OPITIM,R0       ;GET OPIMX TO CALCULATE TIME EXPIRED
017546  052711  000101                  BIS     #101,(R1)       ;ENABLE CLOCK
017552  005737  002664          40$:    TST     OPITIM          ;COUNT EXPIRED?
017556  001446                          BEQ     4$              ;BRANCH - IF YES
017560  005737  002256                  TST     INTFLG          ;INTERRUPT OCCURED?
017564  001772                          BEQ     40$             ;BRANCH - IF NO
```

```
••TEST 7•• - CHECK OPI TIME WITH HDR NT FND

 3029 017566  005437  002664                 NEG     OPITIM          ;GET NEGATIVE OF FACTOR FOR SUBTRACTION
 3030 017572  060037  002664                 ADD     R0,OPITIM       ;SUBTRACT PASSING TIME FROM ORIGINAL
 3031 017576  013700  002664                 MOV     OPITIM,R0       ;GET TIME EXPIRED
 3032 017602  005737  002656                 TST     SIXTY           ;60 HZ.?
 3033 017606  001405                         BEQ     9$              ;BRANCH - IF NO
 3034 017610  006300                         ASL     R0              ;MULTIPLY BY 16(10)
 3035 017612  006300                         ASL     R0              ;FOR
 3036 017614  006300                         ASL     R0              ;60 HZ.
 3037 017616  006300                         ASL     R0              ;CASE
 3038 017620  000410                         BR      2$              ;EXIT
 3039 017622  006300                 9$:     ASL     R0              ;MULTIPLY BY 20(10)
 3040 017624  006300                         ASL     R0              ;FOR
 3041 017626  006300                         ASL     R0              ;THE
 3042 017630  006300                         ASL     R0              ;50 HZ.
 3043 017632  063700  002664                 ADD     OPITIM,R0       ;CASE
 3044 017636  063700  002664                 ADD     OPITIM,R0       ;STOP HERE
 3045
 3046                                 ;CHECK THAT OPI TIME IS WITHIN LIMITS
 3047
 3048 017642  010037  002302          2$:     MOV     R0,BDDAT        ;SAVE EXPIRED TIME
 3049                                 ;       SETPRI  #PRIO7                          ;JSD REV A
 3050 017646                                  SETPRI  #PRIO6                          ;JSD REV A
      017646  012700  000300                 MOV     #PRIO6,R0
      017652  104441                         TRAP    C$SPRI
 3051 017654  023737  002414  002302         CMP     OPIMX,BDDAT     ;IS IT WITHIN LIMITS
 3052 017662  002404                         BLT     4$              ;NO, REPORT ERROR
 3053 017664  023737  002412  002302         CMP     OPIMN,BDDAT     ;WITHIN LIMITS
 3054 017672  003404                         BLE     5$              ;YES
 3055 017674                          4$:     ERRDF   974.,EM56,ERR13 ;OPI TIMING INCORRECT
      017674  104455                         TRAP    C$ERDF
      017676  001716                         .WORD   974
      017700  007033                         .WORD   EM56
      017702  010346                         .WORD   ERR13
 3056 017704                          5$:     CLRVEC  BVEC            ;CLEAR PRESENT VECTOR
      017704  013700  002366                 MOV     BVEC,R0
      017710  104436                         TRAP    C$CVEC
 3057 017712                                  SETVEC  BVEC,#INTSRV,#340       ;SET IN OLD VECTOR
      017712  012746  000340                 MOV     #340,-(SP)
      017716  012746  014464                 MOV     #INTSRV,-(SP)
      017722  013746  002366                 MOV     BVEC,-(SP)
      017726  012746  000003                 MOV     #3,-(SP)
      017732  104437                         TRAP    C$SVEC
      017734  062706  000010                 ADD     #10,SP
 3058 017740                                  ENDSEG                  ;••END OF SEGMENT••
      017740                          10000$:
      017740  104405                         TRAP    C$ESEG
 3059 017742  005037  002652          8$:     CLR     XITFLG          ;INIT EXIT FLAG
 3060 017746  005037  002656                 CLR     SIXTY           ;INIT 60 HZ. FLAG
 3061 017752  005037  002660                 CLR     PCLOCK          ;INIT PCLOCK INDICATOR
 3062
 3063 017756                          ENDTST                          ;••END OF TEST••
      017756                          L10041:
      017756  104401                         TRAP    C$ETST
 3064
 3065                                 .SBTTL  ••TEST 8•• - MULTIPLE SECTOR TRANSFER ON WRITE
 3066
 3067 017760                          BGNTST                                  ;••START OF TEST••
```

TEST 8 - MULTIPLE SECTOR TRANSFER ON WRITE

```
3068
3069 017760                            STARS
                                       ;;*************************************************************************
3070                                   ;CHECK FOR MULTIPLE SECTOR TRANSFER ON WRITE. THIS TEST CHECKS
3071                                   ;THAT TWO SECTORS CAN BE SUCCESSFULLY WRITTEN.  WE LOAD
3072                                   ;A WORD COUNT OF 129 WORDS (ONE SECTOR + 1 WORD) STARTING AT
3073                                   ;SECTOR 0 THRU SECTOR 37 AND VERIFY THAT THE RLDA DOES
3074                                   ;A DOUBLE INCREMENT EACH TIME.
3075 017760                            STARS
                                       ;;*************************************************************************
3076
3077 017760  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3078 017764                                    CKERFG                  ;HEADS GO HOME OKAY
     017772  104432                            TRAP    C$EXIT
     017774  000152                            .WORD   L10042-.
3079
3080 017776  005037  002272                    CLR     TMP0            ;CLEAR TEMP LOCATIONS
3081 020002  005037  002274                    CLR     TMP1
3082
3083 020006                                    BGNSEG                          ;**START OF SEGMENT**
     020006  104404                            TRAP    C$BSEG
3084
3085 020010  013737  002274  002300  1$:      MOV     TMP1,GDDAT      ;GET CYLINDER
3086 020016  053737  002272  002300            BIS     TMP0,GDDAT      ;GET SECTOR
3087 020024  013777  002300  162326            MOV     GDDAT,@RLDA     ;SET DISK ADDRESS-SECTOR 0
3088 020032  062737  000002  002300            ADD     #2,GDDAT        ;SET EXPECTED + 2
3089 020040  012777  003426  162310            MOV     #BUF,@RLBA      ;SET BUS ADDRESS
3090 020046  012777  177577  162306            MOV     #-129.,@RLMP    ;WORD COUNT-SECTOR+1 WORD
3091
3092 020054  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3093 020060  000012                            WRITE                   ;WRITE
3094 020062  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY?
3095 020066                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020066  104410                            TRAP    C$ESCAPE
     020070  000054                            .WORD   10000$-.
3096
3097 020072  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3098 020076                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020076  104410                            TRAP    C$ESCAPE
     020100  000044                            .WORD   10000$-.
3099
3100 020102  013737  002344  002302            MOV     E.DA,BDDAT      ;READ DISK ADDRESS
3101 020110  023737  002302  002300            CMP     BDDAT,GDDAT     ;IS DISK ADDRESS CORRECT
3102 020116  001404                            BEQ     2$              ;YES, BRANCH  NO, REPORT ERROR
3103
3104 020120                                    ERRDF   7.,EM22,ERR4    ;DISK ADDRESS NOT CORRECT
     020120  104455                            TRAP    C$ERDF
     020122  000007                            .WORD   7
     020124  005461                            .WORD   EM22
     020126  007654                            .WORD   ERR4
3105
3106 020130                            2$:
3107
3108 020130  005237  002272                    INC     TMP0            ;NEXT SECTOR
3109 020134  022737  000046  002272            CMP     #46,TMP0        ;AT END?
3110 020142  001322                            BNE     1$              ;NO, GO BACK
3111
```

```
**TEST 8** - MULTIPLE SECTOR TRANSFER ON WRITE

3112 020144                                     ENDSEG                          ;%%END OF SEGMENT%%
     020144                             10000$:
     020144  104405                             TRAP    C$ESEG
3113 020146                             ENDTST                                  ;**END OF TEST**
     020146                             L10042:
     020146  104401                             TRAP    C$ETST
3114
3115                                    .SBTTL  **TEST 9** - CHECK DIRECTION OF WRITE NPR
3116
3117 020150                             BGNTST                                  ;**START OF TEST**
3118
3119 020150                             STARS
                                        ;;**********************************************************************
3120                                    ;VERIFY THAT A WRITE IS WRITING NOT READING.  WE WRITE A
3121                                    ;KNOWN PATTERN IN "BUF" (128 WORD), WE THEN ISSUE A WRITE.
3122                                    ;ONCE THE WRITE IS FINISHED WE CHECK THAT "BUF" IS INTACT.
3123                                    ;THIS IS DONE TO PROVE THAT THE NPR IS GOING THE RIGHT
3124                                    ;WAY.
3125 020150                             STARS
                                        ;;**********************************************************************
3126
3127 020150  004737  015764                     JSR     PC,@DHOME       ;HEADS OVER TRACK 0
3128 020154                                     CKERFG                  ;HEADS GO HOME OKAY
     020162  104432                             TRAP    C$EXIT
     020164  000160                             .WORD   L10043-.
3129
3130 020166                                     BGNSEG                          ;%%START OF SEGMENT%%
     020166  104404                             TRAP    C$BSEG
3131
3132 020170                             2$:
3133 020170  012702  003426                     MOV     #BUF,R2         ;WRITE BUFFER FOR WRITE OPERATION
3134 020174  012701  000200                     MOV     #128.,R1        ;ONE SECTOR'S WORTH
3135 020200  012722  125252             3$:     MOV     #125252,(R2)+   ;WRITE BUFFER
3136 020204  005301                             DEC     R1              ;DONE?
3137 020206  001374                             BNE     3$              ;NO, GO BACK
3138
3139 020210  005077  162144                     CLR     @RLDA           ;LOAD DISK ADDRESS
3140 020214  012777  177600  162140             MOV     #-128.,@RLMP    ;WORD COUNT
3141 020222  012777  003426  162126             MOV     #BUF,@RLBA      ;BUS ADDRESS
3142 020230  004537  015054                     JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3143 020234  000012                             WRITE                   ;WRITE SOME DATA
3144 020236  004537  015700                     JSR     R5,WTCRDY       ;WAIT FOR IT TO FINISH
3145 020242                                     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020242  104410                             TRAP    C$ESCAPE
     020244  000076                             .WORD   10000$-.
3146
3147 020246  004537  014612                     JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3148 020252                                     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020252  104410                             TRAP    C$ESCAPE
     020254  000066                             .WORD   10000$-.
3149
3150 020256  012702  003426                     MOV     #BUF,R2         ;SET UP TO CHECK BUFFER
3151 020262  012701  000200                     MOV     #128.,R1        ;CHECK 128 WORDS
3152
3153 020266                                     BGNSEG                          ;%%START OF SEGMENT%%
     020266  104404                             TRAP    C$BSEG
3154
```

```
**TEST 9** - CHECK DIRECTION OF WRITE NPR

3155 020270  012737  125252  002300          MOV     #125252,GDDAT   ;DATA SHOULD BE 125252
3156 020276  011237  002302          4$:     MOV     (R2),BDDAT      ;LOAD DATA INTO BDDAT
3157 020302  023737  002300  002302          CMP     GDDAT,BDDAT     ;IS IT OKAY?
3158 020310  001406                          BEQ     5$              ;YES, CONTINUE
3159
3160 020312  010237  002274                  MOV     R2,TMP1 ;LOAD MEMORY LOCATION OF FAILURE
3161 020316                                  ERRDF   8.,EM26,ERR8
     020316  104455                          TRAP    C$ERDF
     020320  000010                          .WORD   8
     020322  005710                          .WORD   EM26
     020324  010030                          .WORD   ERR8
3162
3163 020326                          5$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020326  104410                          TRAP    C$ESCAPE
     020330  000010                          .WORD   10001$-.
3164 020332  005722                  6$:     TST     (R2)+           ;NEXT!
3165 020334  005301                          DEC     R1              ;DONE?
3166 020336  001357                          BNE     4$              ;NO, GO BACK
3167
3168 020340                                  ENDSEG                          ;**END OF SEGMENT**
     020340                          10001$:
     020340  104405                          TRAP    C$ESEG
3169 020342                                  ENDSEG                          ;**END OF SEGMENT**
     020342                          10000$:
     020342  104405                          TRAP    C$ESEG
3170 020344                          ENDTST                                  ;**END OF TEST**
     020344                          L10043:
     020344  104401                          TRAP    C$ETST
3171
3172                                 .SBTTL  **TEST 10** - CHECK FULL RLBA INCREMENT
3173
3174 020346                          BGNTST                                  ;**START OF TEST**
3175
3176 020346                          STARS
                                     ;;**********************************************************************
3177                                 ;TEST THAT THE RLBA WILL INCREMENT, WE DO NOT DO A FULL 16
3178                                 ;BIT INCREMENT WE CHECK THAT EACH BIT WILL TOGGLE 0 TO 1
3179                                 ;AND 1 TO 0.  WE DO CHECK ALL BITS EVEN IF ALL MEMORY
3180                                 ;IS NOT AVAILABLE.  (WE IGNORE NON-EXISTANT MEMORY ERRORS).
3181                                 ;WE USE THE SAME DISK ADDRESS (RANDOM) AND A 1 WORD TRANSFER.
3182 020346                          STARS
                                     ;;**********************************************************************
3183
3184 020346  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3185 020352                                  CKERFG                  ;HEADS GO HOME OKAY
     020360  104432                          TRAP    C$EXIT
     020362  000134                          .WORD   L10044-.
3186
3187 020364  005037  002274                  CLR     TMP1            ;CLEAR LOCATION
3188
3189 020370                                  BGNSEG                          ;**START OF SEGMENT**
     020370  104404                          TRAP    C$BSEG
3190
3191 020372                          3$:
3192 020372  012777  177777  161762          MOV     #-1,@RLMP       ;ONLY ONE (1) WORD
3193 020400  005077  161754                  CLR     @RLDA           ;LOAD DISK ADDRESS
3194 020404  013777  002274  161744          MOV     TMP1,@RLBA      ;BUS ADDRESS
```

```
**TEST 10** - CHECK FULL RLBA INCREMENT

3195
3196 020412  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3197 020416  000012                             WRITE
3198 020420  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR WRITE TO FINISH
3199 020424                                     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020424  104410                             TRAP    C$ESCAPE
     020426  000066                             .WORD   10000$-.
3200
3201 020430  013737  002274  002300  4$:       MOV     TMP1,GDDAT      ;SET UP EXPECTED RLBA
3202 020436  062737  000002  002300            ADD     #2,GDDAT        ;PREVIOUS RLBA+2
3203 020444  013737  002342  002302            MOV     E.BA,BDDAT      ;READ RLBA
3204 020452  023737  002300  002302            CMP     GDDAT,BDDAT     ;WAS IT UPDATED PROPERLY?
3205 020460  001404                            BEQ     5$              ;YES, CONTINUE
3206
3207 020462                                     ERRDF   9.,EM30,ERR4    ;BA INCREMENT ERROR
     020462  104455                             TRAP    C$ERDF
     020464  000011                             .WORD   9
     020466  006005                             .WORD   EM30
     020470  007654                             .WORD   ERR4
3208 020472                             5$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020472  104410                             TRAP    C$ESCAPE
     020474  000020                             .WORD   10000$-.
3209
3210 020476  006337  002274                    ASL     TMP1            ;NEXT PATTERN TO TEST RLBA
3211 020502  103404                            BCS     6$              ;DONE?
3212 020504  052737  000002  002274            BIS     #BIT1,TMP1      ;NO, SET IN BIT 1
3213 020512  000727                            BR      3$              ;GO CHECK NEXT.
3214
3215 020514                             6$:                             ;END TEST
3216
3217 020514                                     ENDSEG                          ;**END OF SEGMENT**
     020514                             10000$:
     020514  104405                             TRAP    C$ESEG
3218 020516                             ENDTST                          ;**END OF TEST**
     020516                             L10044:
     020516  104401                             TRAP    C$ETST
3219
3220                                    .SBTTL  **TEST 11** - BA BIT 16 INCREMENT
3221
3222 020520                             BGNTST                          ;**START OF TEST**
3223
3224 020520                             STARS
                                        ;;*************************************************************************
3225                                    ;CHECK THAT BA BIT 16 WILL INCREMENT.  WE WILL LOAD THE
3226                                    ;RLBA WITH 177776 AND ISSUE A ONE WORD WRITE WE THEN
3227                                    ;CHECK BA BIT 16 TO SET, BA 17 TO STAY A 0 AND THE RLBA
3228                                    ;TO GO TO ZERO
3229 020520                             STARS
                                        ;;*************************************************************************
3230
3231 020520  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3232 020524                                     CKERFG                  ;HEADS GO HOME OKAY
     020532  104432                             TRAP    C$EXIT
     020534  000160                             .WORD   L10045-.
3233
3234 020536                                     BGNSEG                          ;**START OF SEGMENT**
     020536  104404                             TRAP    C$BSEG
```

TEST 11 - BA BIT 16 INCREMENT

```
3235
3236 020540                                2$:
3237 020540  012777  177776  161610              MOV     #177776,@RLBA   ;SET MAX BA TO INC. BA16
3238 020546  005037  002374                      CLR     XMEM            ;WE DON'T WANT TO LOAD ANY EA
3239 020552  012777  177777  161602              MOV     #-1,@RLMP       ;ONE WORD TRANSFER
3240 020560  005077  161574                      CLR     @RLDA
3241 020564  004537  015054                      JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3242 020570  000012                              WRITE
3243 020572  004537  015700                      JSR     R5,WTCRDY       ;WAIT FOR WRITE TO FINISH
3244 020576                                      ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020576  104410                              TRAP    C$ESCAPE
     020600  000112                              .WORD   10000$-.
3245 020602  032737  020000  002340              BIT     #NXM,E.CS       ;NON-EXISTANT MEMORY ERROR?
3246 020610  001002                              BNE     3$              ;YES, CONTINUE
3247
3248 020612  004537  014612                      JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3249 020616                                3$:   ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     020616  104410                              TRAP    C$ESCAPE
     020620  000072                              .WORD   10000$-.
3250
3251 020622  032737  000020  002340              BIT     #BA16,E.CS      ;DID BA16 SET?
3252 020630  001004                              BNE     4$              ;YES, CONTINUE
3253
3254 020632                                      ERRDF   10.,EM31,ERR0   ;BA 16 DID NOT INCREMENT
     020632  104455                              TRAP    C$ERDF
     020634  000012                              .WORD   10
     020636  006040                              .WORD   EM31
     020640  007510                              .WORD   ERR0
3255
3256 020642                                4$:   CKLOOP
     020642  104406                              TRAP    C$CLP1
3257
3258 020644  032737  000040  002340              BIT     #BA17,E.CS      ;DID BA17 SET ALSO?
3259 020652  001404                              BEQ     5$              ;NO, GOOD CONTINUE
3260
3261 020654                                      ERRDF   11.,EM32,ERR0   ;BA 17 GOT CARRIED AWAY
     020654  104455                              TRAP    C$ERDF
     020656  000013                              .WORD   11
     020660  006076                              .WORD   EM32
     020662  007510                              .WORD   ERR0
3262
3263 020664                                5$:   CKLOOP
     020664  104406                              TRAP    C$CLP1
3264
3265 020666  005037  002300                      CLR     @DDAT           ;CHECK THAT BA15-BA0 IS CLEAR
3266 020672  013737  002342  002302              MOV     E.BA,@DDAT      ;READ BA
3267 020700  001404                              BEQ     6$              ;IS BA ZERO?
3268 020702                                      ERRDF   12.,EM33,ERR4   ;BA SHOULD BE ZERO
     020702  104455                              TRAP    C$ERDF
     020704  000014                              .WORD   12
     020706  006135                              .WORD   EM33
     020710  007654                              .WORD   ERR4
3269
3270 020712                                6$:                           ;
3271
3272 020712                                      ENDSEG                          ;**END OF SEGMENT**
     020712                                10000$:
```

TEST 11 - BA BIT 16 INCREMENT

```
      020712  104405                          TRAP    C$ESEG
 3273 020714                          ENDTST                          ;**END OF TEST**
      020714                          L10045:
      020714  104401                          TRAP    C$ETST
 3274
 3275                                 .SBTTL  **TEST 12** - BA BIT 17 INCREMENT
 3276
 3277 020716                          BGNTST                          ;**START OF TEST**
 3278
 3279 020716                          STARS
                                      ;;*************************************************************************
 3280                                 ;CHECK THAT BA BIT 17 WILL INCREMENT.  WE WILL LOAD THE
 3281                                 ;RLBA WITH 177776 AND BA 16 SET, WE WILLISSUE A ONE WORD
 3282                                 ;WRITE.  WE THEN CHECK BA17 TO SET, BA16 TO CLEAR AND
 3283                                 ;BA15 - BA0 TO CLEAR.
 3284 020716                          STARS
                                      ;;*************************************************************************
 3285
 3286 020716  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 3287 020722                                  CKERFG                  ;HEADS GO HOME OKAY
      020730  104432                          TRAP    C$EXIT
      020732  000162                          .WORD   L10046-.
 3288
 3289 020734                                  BGNSEG                          ;%%START OF SEGMENT%%
      020734  104404                          TRAP    C$BSEG
 3290
 3291 020736                          2$:
 3292 020736  012777  177776  161412          MOV     #177776,@RLBA   ;SET MAX BA TO INC. BA16
 3293 020744  012737  000020  002374          MOV     #BA16,XMEM      ;SET BA16 IN RLCS
 3294 020752  012777  177777  161402          MOV     #-1,@RLMP       ;ONE WORD TRANSFER
 3295 020760  005077  161374                  CLR     @RLDA
 3296 020764  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 3297 020770  000012                          WRITE
 3298 020772  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR WRITE TO FINISH
 3299 020776                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      020776  104410                          TRAP    C$ESCAPE
      021000  000112                          .WORD   10000$-.
 3300 021002  032737  020000  002340          BIT     #NXM,E.CS       ;NON-EXISTANT MEMORY ERROR?
 3301 021010  001002                          BNE     3$              ;YES, CONTINUE
 3302
 3303 021012  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 3304 021016                          3$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      021016  104410                          TRAP    C$ESCAPE
      021020  000072                          .WORD   10000$-.
 3305
 3306 021022  032737  000040  002340          BIT     #BA17,E.CS      ;DID BA17 SET?
 3307 021030  001004                          BNE     4$              ;YES, CONTINUE
 3308
 3309 021032                                  ERRDF   13.,EM34,ERRO   ;BA 17 DID NOT SET
      021032  104455                          TRAP    C$ERDF
      021034  000015                          .WORD   13
      021036  006171                          .WORD   EM34
      021040  007510                          .WORD   ERRO
 3310
 3311 021042                          4$:     CKLOOP
      021042  104406                          TRAP    C$CLP1
 3312
```

```
**TEST 12** - BA BIT 17 INCREMENT

3313 021044  032737  000020  002340          BIT     #BA16,E.CS      ;DID BA16 SET ALSO?
3314 021052  001404                          BEQ     5$              ;NO, GOOD CONTINUE
3315
3316 021054                                  ERRDF   14.,EM35,ERR0   ;BA 16 DIDN'T KNOW WHEN TO QUIT.
     021054  104455                          TRAP    C$ERDF
     021056  000016                          .WORD   14
     021060  006227                          .WORD   EM35
     021062  007510                          .WORD   ERR0
3317 021064                          5$:     CKLOOP
     021064  104406                          TRAP    C$CLP1
3318
3319 021066  005037  002300                  CLR     GDDAT           ;CHECK THAT BA15-BA0 IS CLEAR
3320 021072  013737  002342  002302          MOV     E.BA,BDDAT      ;READ BA
3321 021100  001404                          BEQ     6$              ;IS BA ZERO?
3322 021102                                  ERRDF   15.,EM36,ERR4   ;BA SHOULD BE ZERO
     021102  104455                          TRAP    C$ERDF
     021104  000017                          .WORD   15
     021106  006265                          .WORD   EM36
     021110  007654                          .WORD   ERR4
3323
3324 021112                          6$:                             ;
3325
3326 021112                                  ENDSEG                          ;**END OF SEGMENT**
     021112                          10000$:
     021112  104405                          TRAP    C$ESEG
3327 021114                          ENDTST                          ;**END OF TEST**
     021114                          L10046:
     021114  104401                          TRAP    C$ETST
3328
3329                                 .SBTTL  **TEST 13** - READ FUNCTION
3330
3331 021116                          BGNTST                          ;**START OF TEST**
3332
3333 021116                          STARS
                                     ;;**************************************************************
3334                                 ;CHECK OF THE READ FUNCTION.  WE WILL FIRST DO A READ
3335                                 ;HEADER TO FIND OUT WHERE WE ARE AND THEN ISSUE
3336                                 ;A FULL SECTOR READ, WAIT FOR READY AND CHECK FOR
3337                                 ;ANY ERRORS
3338 021116                          STARS
                                     ;;**************************************************************
3339
3340 021116  004737  015764                  JSR     PC,MDHOME       ;HEADS OVER TRACK 0
3341 021122                                  CKERFG                  ;HEADS GO HOME OKAY
     021130  104432                          TRAP    C$EXIT
     021132  000064                          .WORD   L10047-.
3342
3343 021134                                  BGNSEG                          ;**START OF SEGMENT**
     021134  104404                          TRAP    C$BSEG
3344
3345 021136  012737  001750  002272          MOV     #1000.,TMP0
3346 021144  005077  161210          1$:     CLR     @RLDA           ;LOAD DISK ADDRESS
3347 021150  012777  177600  161204          MOV     #-128.,@RLMP    ;SET WORD LENGTH
3348 021156  012777  003426  161172          MOV     #BUF,@RLBA      ;SET BUS ADDRESS
3349
3350 021164  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3351 021170  000014                          READ                    ;READ
```

```
**TEST 13** - READ FUNCTION

3352 021172  004537  015700                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3353 021176                                 ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     021176  104410                         TRAP    C$ESCAPE
     021200  000014                         .WORD   10000$-.
3354
3355 021202  004537  014612                JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3356
3357 021206  005337  002272                DEC     TMP0
3358 021212  001354                        BNE     1$
3359 021214                                 ENDSEG                          ;**END OF SEGMENT**
     021214                         10000$:
     021214  104405                         TRAP    C$ESEG
3360 021216                         ENDTST                                  ;**END OF TEST**
     021216                         L10047:
     021216  104401                         TRAP    C$ETST
3361
3362                                .SBTTL  **TEST 14** - READ FUNCTION INTERRUPT
3363
3364 021220                         BGNTST                                  ;**START OF TEST**
3365
3366 021220                         STARS
                                    ;;**************************************************************************
3367                                ;CHECK OF THE READ FUNCTION UNDER INTERRUPT CONTROL, WE WILL
3368                                ;ISSUE A READ HEADER TO GET POSITION AND THEN READ
3369                                ;A FULL SECTOR WAITING FOR THE INTERRUPT.  CHECK FOR
3370                                ;ERRORS ON INTERRUPT.
3371 021220                         STARS
                                    ;;**************************************************************************
3372
3373 021220  004737  015764                JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3374 021224                                 CKERFG                  ;HEADS GO HOME OKAY
     021232  104432                         TRAP    C$EXIT
     021234  000106                         .WORD   L10050-.
3375
3376 021236                                 BGNSEG                          ;**START OF SEGMENT**
     021236  104404                         TRAP    C$BSEG
3377
3378 021240  005037  002256                CLR     INTFLG          ;CLEAR INTERRUPT INDICATOR
3379 021244  005077  161110                CLR     @RLDA           ;SET DISK ADDRESS
3380 021250  012777  177600  161104        MOV     #-128.,@RLMP    ;SET UP WORD COUNT
3381 021256  012777  003426  161072        MOV     #BUF,@RLBA      ;SET UP BUS ADDRESS
3382
3383 021264                                 SETPRI  #PRI00          ;PRIORITY TO 0
     021264  012700  000000                MOV     #PRI00,R0
     021270  104441                         TRAP    C$SPRI
3384 021272  004537  015054                JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3385 021276  000114                        READ!INTEN              ;READ UNDER INTERRUPT
3386 021300  004537  015700                JSR     R5,WTCRDY       ;WAIT FOR INTERRUPT
3387 021304                                 CKLOOP
     021304  104406                         TRAP    C$CLP1
3388                                ;       SETPRI  #PRI07          ;PRIORITY TO 7                  ;JSD REV A
3389 021306                                 SETPRI  #PRI06          ;PRIORITY TO 6                  ;JSD REV A
     021306  012700  000300                MOV     #PRI06,R0
     021312  104441                         TRAP    C$SPRI
3390
3391 021314  005737  002256                TST     INTFLG          ;DID INTERRUPT OCCUR?
3392 021320  001004                        BNE     1$              ;YES-BRANCH NO-REPORT
```

TEST 14 - READ FUNCTION INTERRUPT

```
3393
3394 021322                            ERRDF   19.,EM4,ERRO    ;READ DID NOT INTERRUPT
     021322  104455                    TRAP    C$ERDF
     021324  000023                    .WORD   19
     021326  004712                    .WORD   EM4
     021330  007510                    .WORD   ERRO
3395 021332                     1$:     CKLOOP                  ;CHECK FOR LOOP
     021332  104406                    TRAP    C$CLP1
3396
3397 021334  004537  014612            JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3398
3399 021340                            ENDSEG                          ;**END OF SEGMENT**
     021340                     10000$:
     021340  104405                    TRAP    C$ESEG
3400 021342                     ENDTST                                 ;**END OF TEST**
     021342                     L10050:
     021342  104401                    TRAP    C$ETST
3401
3402                            .SBTTL  **TEST 15** - CHECK READ NPR DIRECTION
3403
3404 021344                     BGNTST                                 ;**START OF TEST**
3405
3406 021344                     STARS
                                ;;*************************************************************************
3407                            ;CHECK THAT THE READ FUNCTION ACTUALLY READS (INTO MEMORY)
3408                            ;WE WILL WRITE A PATTERN INTO MEMORY AND THEN ISSUE
3409                            ;A READ TO OVERLAY THAT PATTERN.  AFTER THE READ
3410                            ;WE CHECK TO SEE IF THE WRITTEN PATTERN HAS CHANGED.
3411                            ;IF NOT WE ISSUE IT AGAIN AT THE SAME SECTION AFTER
3412                            ;HAVING MODIFIED OUR PATTERN IN MEMORY (SINCE THERE IS
3413                            ;ONE CHANCE THAT THE DISK COULD HAVE OUR PATTERN).  AFTER
3414                            ;THE SECOND READ WE CHECK THE BUFFER AGAIN.  IF IT'S
3415                            ;NO CHANGED WE REPORT AN ERROR
3416 021344                     STARS
                                ;;*************************************************************************
3417
3418 021344  004737  015764            JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3419 021350                            CKERFG                  ;HEADS GO HOME OKAY
     021356  104432                    TRAP    C$EXIT
     021360  000156                    .WORD   L10051-.
3420
3421 021362                            BGNSEG                          ;**START OF SEGMENT**
     021362  104404                    TRAP    C$BSEG
3422
3423 021364  012737  123456  002272            MOV     #123456,TMP0    ;SET PATTERN TO WRITE
3424 021372  005037  002274            CLR     TMP1            ;CLEAR PASS INDICATOR
3425 021376  012700  003426     1$:    MOV     #BUF,R0         ;SET UP BUFFER BEGINNING
3426 021402  012701  000200            MOV     #128.,R1
3427 021406  013720  002272     2$:    MOV     TMP0,(R0)+      ;WRITE BUFFER
3428 021412  005301                    DEC     R1              ;DONE??
3429 021414  001374                    BNE     2$              ;NO, GO BACK
3430 021416  005077  160736            CLR     @RLDA           ;LOAD DISK ADDRESS
3431 021422  012777  177600  160732    MOV     #-128.,@RLMP    ;SET WORD COUNT
3432 021430  012777  003426  160720    MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
3433 021436  012737  003426  002300    MOV     #BUF,GDDAT      ;FOR ERROR PRINT
3434
3435 021444  004537  015054            JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
```

```
**TEST 15** - CHECK READ NPR DIRECTION

3436 021450  000014                        READ                    ;READ
3437 021452  004537  015700                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3438 021456                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     021456  104410                        TRAP    C$ESCAPE
     021460  000054                        .WORD   10000$-.
3439
3440 021462  004537  014612                JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3441 021466                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     021466  104410                        TRAP    C$ESCAPE
     021470  000044                        .WORD   10000$-.
3442
3443 021472  012702  003426                MOV     #BUF,R2         ;SET TO START COMPARING DATA
3444 021476  022237  002272          4$:   CMP     (R2)+,TMP0      ;DID DATA CHANGE?
3445 021502  001014                        BNE     6$              ;YES, CHECK FOR END
3446                                       ;
3447
3448                                       ;DATA DIDN'T CHANGE, CHECK
3449 021504  005737  002274                TST     TMP1            ;IF 1ST OR 2ND TIME?
3450 021510  001005                        BNE     5$              ;2ND-REPORT 1ST-TRY AGAIN
3451
3452 021512  005237  002274                INC     TMP1            ;INC PASS COUNT
3453 021516  005137  002272                COM     TMP0            ;COMPLIMENT PATTERN
3454 021522  000725                        BR      1$              ;GO DO IT AGAIN
3455
3456 021524                          5$:   ERRDF   20.,EM5,ERR9    ;READ DID NOT MODIFY MEMORY
     021524  104455                        TRAP    C$ERDF
     021526  000024                        .WORD   20
     021530  004735                        .WORD   EM5
     021532  010102                        .WORD   ERR9
3457
3458 021534                          6$:
3459
3460 021534                                ENDSEG                          ;**END OF SEGMENT**
     021534                          10000$:
     021534  104405                        TRAP    C$ESEG
3461 021536                          ENDTST                            ;**END OF TEST**
     021536                          L10051:
     021536  104401                        TRAP    C$ETST
3462
3463                                 .SBTTL  **TEST 16** - PROPER INCREMENT OF RLBA ON READ
3464
3465 021540                          BGNTST                            ;**START OF TEST**
3466
3467 021540                          STARS
                                     ;;**********************************************************************
3468                                 ;CHECK THAT THE RLBA WILL INCREMENT WITH THE READ
3469                                 ;THE RLBA SHOULD CONTAIN "BUF +256."  AFTER A FULL SECTOR
3470                                 ;READ.
3471 021540                          STARS
                                     ;;**********************************************************************
3472
3473 021540  004737  015764                JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3474 021544                                CKERFG                  ;HEADS GO HOME OKAY
     021552  104432                        TRAP    C$EXIT
     021554  000116                        .WORD   L10052-.
3475
3476 021556                                BGNSEG                          ;**START OF SEGMENT**
```

```
**TEST 16** - PROPER INCREMENT OF RLBA ON READ

        021556  104404                          TRAP    C$BSEG
 3477
 3478 021560  005077  160574                    CLR     @RLDA           ;SET UP DISK ADDRESS
 3479 021564  012777  003426  160564            MOV     #BUF,@RLBA      ;SET UP BUS ADDRESS
 3480 021572  012777  177600  160562            MOV     #-128.,@RLMP    ;WORD COUNT
 3481 021600  012737  003426  002300            MOV     #BUF,GDDAT      ;FORM EXPECTED BUS ADDRESS
 3482 021606  062737  000400  002300            ADD     #256.,GDDAT     ;AFTER READ
 3483
 3484 021614  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 3485 021620  000014                            READ                    ;READ
 3486 021622  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
 3487 021626                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
        021626  104410                          TRAP    C$ESCAPE
        021630  000040                          .WORD   10000$-.
 3488
 3489 021632  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 3490 021636                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
        021636  104410                          TRAP    C$ESCAPE
        021640  000030                          .WORD   10000$-.
 3491 021642  013737  002342  002302            MOV     E.BA,BDDAT      ;READ 'RLBA' FOR PRESENT ADDRESS
 3492 021650  023737  002302  002300            CMP     BDDAT,GDDAT     ;DID 'BA' INCREMENT PROPERLY?
 3493 021656  001404                            BEQ     1$              ;YES, CONTINUE
 3494
 3495 021660                                    ERRDF   21.,EM6,ERR4    ;BA DID NOT INCREMENT PROPERLY
        021660  104455                          TRAP    C$ERDF
        021662  000025                          .WORD   21
        021664  004763                          .WORD   EM6
        021666  007654                          .WORD   ERR4
 3496
 3497 021670                            1$:
 3498
 3499 021670                                    ENDSEG                          ;**END OF SEGMENT**
        021670
        021670  104405                          TRAP    C$ESEG
 3500 021672                            ENDTST                                  ;**END OF TEST**
        021672                          L10052:
        021672  104401                          TRAP    C$ETST
 3501
 3502                                   .SBTTL  **TEST 17** - PROPER INCREMENT OF RLDA ON READ
 3503
 3504 021674                            BGNTST                                  ;**START OF TEST**
 3505
 3506 021674                            STARS
                                        ;;**********************************************************************
 3507                                   ;CHECK THAT THE RLDA INCREMENTS BY ONE AFTER A
 3508                                   ;FULL SECTOR READ, WE FIRST READ A HEADER TO FIND
 3509                                   ;OUT WHERE WE ARE, THEN ISSUE A READ AFTER
 3510                                   ;THE READ THE RLDA SHOULD BE RLDA (START) + 1
 3511 021674                            STARS
                                        ;;**********************************************************************
 3512
 3513 021674  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 3514 021700                                    CKERFG                  ;HEADS GO HOME OKAY
        021706  104432                          TRAP    C$EXIT
        021710  000114                          .WORD   L10053-.
 3515
 3516 021712                                    BGNSEG                          ;**START OF SEGMENT**
```

```
**TEST 17** - PROPER INCREMENT OF RLDA ON READ

        021712  104404                          TRAP    C$BSEG
3517
3518    021714  005037  002300                  CLR     GDDAT
3519    021720  013777  002300  160432          MOV     GDDAT,@RLDA     ;SETUP DISK ADDRESS
3520    021726  005237  002300                  INC     GDDAT           ;CREATE EXPECTED SECTOR
3521    021732  012777  177600  160422          MOV     #-128.,@RLMP    ;WORD COUNT
3522    021740  012777  003426  160410          MOV     #BUF,@RLBA      ;SETUP BUS ADDRESS
3523
3524    021746  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3525    021752  000014                          READ                    ;READ
3526    021754  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3527    021760                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
        021760  104410                          TRAP    C$ESCAPE
        021762  000040                          .WORD   10000$-.
3528
3529    021764  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3530    021770                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
        021770  104410                          TRAP    C$ESCAPE
        021772  000030                          .WORD   10000$-.
3531
3532    021774  013737  002344  002302          MOV     E.DA,BDDAT      ;READ DISK ADDRESS
3533    022002  023737  002300  002302          CMP     GDDAT,BDDAT     ;DID SECTOR INCREMENT PROPERLY
3534    022010  001404                          BEQ     1$              ;YES, BRANCH  NO, REPORT ERROR
3535
3536    022012                                  ERRDF   22.,EM7,ERR4    ;DISK ADDRESS DID NOT INCREMENT
        022012  104455                          TRAP    C$ERDF
        022014  000026                          .WORD   22
        022016  005017                          .WORD   EM7
        022020  007654                          .WORD   ERR4
3537
3538    022022                          1$:
3539
3540    022022                                  ENDSEG                          ;**END OF SEGMENT**
        022022                          10000$:
        022022  104405                          TRAP    C$ESEG
3541    022024                          ENDTST                          ;**END OF TEST**
        022024                          L10053:
        022024  104401                          TRAP    C$ETST
3542
3543                                    .SBTTL  **TEST 18** - FORCE HEADER NOT FOUND WITH READ
3544
3545    022026                          BGNTST                          ;**START OF TEST**
3546
3547    022026                          STARS
                                        ;;*************************************************************************
3548                                    ;FORCE HEADER NOT FOUND ERROR TO OCCUR.  THIS IS DONE
3549                                    ;BY SETTING SECTOR 40 OF THE RLDA AND ISSUING A
3550                                    ;READ.  SECTOR 40 DOES NOT EXIST ON THE RL01 PACK
3551                                    ;THEREFORE HDR NT FOUND SHOULD SET.
3552    022026                          STARS
                                        ;;*************************************************************************
3553
3554    022026  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3555    022032                                  CKERFG                  ;HEADS GO HOME OKAY
        022040  104432                          TRAP    C$EXIT
        022042  000102                          .WORD   L10054-.
3556
```

TEST 18 - FORCE HEADER NOT FOUND WITH READ

```
3557 022044                                    BGNSEG                          ;**START OF SEGMENT**
     022044  104404                            TRAP    C$BSEG
3558
3559 022046  012777  000050  160304            MOV     #40.,@RLDA      ;INSURE NOT TO FIND HEADER BY
3560 022054  012777  003426  160274            MOV     #BUF,@RLBA      ;SETTING SECTOR 40 OF CYL. ADDR.
3561 022062  012777  177777  160272            MOV     #-1,@RLMP       ;WORD COUNT
3562
3563 022070  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3564 022074  000014                            READ                    ;READ
3565 022076  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3566 022102                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022102  104410                            TRAP    C$ESCAPE
     022104  000036                            .WORD   10000$-.
3567
3568 022106  013737  002340  002272            MOV     E.CS,TMP0       ;GET RLCS
3569 022114  042737  001777  002272            BIC     #1777,TMP0      ;SAVE ERROR BITS
3570 022122  022737  112000  002272            CMP     #BIT15!BIT12!BIT10,TMP0  ;HDR NOT FOUND SET.
3571 022130  001404                            BEQ     1$              ;YES, CONTINUE
3572
3573 022132                                    ERRDF   23.,EM10,ERR0   ;HEADER NOT FOUND WOULD NOT SET
     022132  104455                            TRAP    C$ERDF
     022134  000027                            .WORD   23
     022136  005064                            .WORD   EM10
     022140  007510                            .WORD   ERR0
3574
3575 022142                            1$:
3576                                   ;
3577
3578 022142                                    ENDSEG                          ;**END OF SEGMENT**
     022142                            10000$:
     022142  104405                            TRAP    C$ESEG
3579 022144                            ENDTST                          ;**END OF TEST**
     022144                            L10054:
     022144  104401                            TRAP    C$ETST
3580
3581                                   .SBTTL  **TEST 19** - FORCE HEADER NOT FOUND WITH READ INTERRUPT
3582
3583 022146                            BGNTST                          ;**START OF TEST**
3584
3585 022146                            STARS
                                       ;;*************************************************************************
3586                                   ;TEST THAT HEADER NOT FOUND ERROR WILL GENERATE AN INTERRUPT
3587                                   ;ON OCCURANCE.  HEADER NOT FOUND WILL BE FORCED BY SETTING
3588                                   ;SECTOR 40 OF RLDA AND ISSUING A READ
3589 022146                            STARS
                                       ;;*************************************************************************
3590
3591 022146  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3592 022152                                    CKERFG                  ;HEADS GO HOME OKAY
     022160  104432                            TRAP    C$EXIT
     022162  000142                            .WORD   L10055-.
3593
3594 022164                                    BGNSEG                          ;**START OF SEGMENT**
     022164  104404                            TRAP    C$BSEG
3595
3596 022166                                    SETPRI  #PRI00
     022166  012700  000000                    MOV     #PRI00,R0
```

```
**TEST 19** - FORCE HEADER NOT FOUND WITH READ INTERRUPT

     022172  104441                          TRAP    C$SPRI
3597 022174  005037  002256                  CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE FLAG
3598 022200  012777  000050  160152          MOV     #40.,@RLDA      ;INSURE NOT TO FIND HEADER BY
3599 022206  012777  003426  160142          MOV     #BUF,@RLBA      ;SETTING SECTOR 40 OF CYL. ADDR.
3600 022214  012777  177777  160140          MOV     #-1,@RLMP       ;WORD COUNT
3601
3602 022222  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3603 022226  000114                          READ!INTEN              ;READ
3604 022230  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3605 022234                                  CKLOOP
     022234  104406                          TRAP    C$CLP1
3606                                 ;       SETPRI  #PRI07                                  ;JSD REV A
3607 022236                                  SETPRI  #PRI06                                  ;JSD REV A
     022236  012700  000300                  MOV     #PRI06,R0
     022242  104441                          TRAP    C$SPRI
3608
3609 022244  005737  002256                  TST     INTFLG          ;DID INTERRUPT OCCUR
3610 022250  001004                          BNE     2$              ;YES
3611
3612 022252                                  ERRDF   24.,EM43,ERR0   ;HNF DID NOT INTERRUPT
     022252  104455                          TRAP    C$ERDF
     022254  000030                          .WORD   24
     022256  006461                          .WORD   EM43
     022260  007510                          .WORD   ERR0
3613
3614 022262                          2$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022262  104410                          TRAP    C$ESCAPE
     022264  000036                          .WORD   10000$-.
3615
3616 022266  013737  002340  002272          MOV     E.CS,TMP0       ;GET RLCS
3617 022274  042737  001777  002272          BIC     #1777,TMP0      ;SAVE ERROR BITS
3618 022302  022737  112000  002272          CMP     #BIT15!BIT12!BIT10,TMP0  ;WDR NOT FOUND SET.
3619 022310  001404                          BEQ     1$              ;YES, CONTINUE
3620
3621 022312                                  ERRDF   25.,EM10,ERR0
     022312  104455                          TRAP    C$ERDF
     022314  000031                          .WORD   25
     022316  005064                          .WORD   EM10
     022320  007510                          .WORD   ERR0
3622                                                                 ;WHEN FORCED
3623 022322                          1$:
3624
3625 022322                                  ENDSEG                                  ;**END OF SEGMENT**
     022322                          10000$:
     022322  104405                          TRAP    C$ESEG
3626 022324                          ENDTST                                          ;**END OF TEST**
     022324                          L10055:
     022324  104401                          TRAP    C$ETST
3627
3628                                 .SBTTL  **TEST 20** - CHECK HEADER COMPARE LOGIC
3629
3630 022326                          BGNTST                                          ;**START OF TEST**
3631
3632 022326                          STARS
                                     ;;**********************************************************************
3633                                 ;CHECK THE HEADER COMPARE LOGIC WORKS.  UP TO THIS POINT WE
3634                                 ;KNOW THAT THE LOGIC FUNCTIONS PROPERLY BUT NOW WE WILL
```

TEST 20 - CHECK HEADER COMPARE LOGIC

```
3635                                    ;CHECK ALL THE BITS IN THE HEADER WORD.  FOUR PATTERNS
3636                                    ;ARE USED A WALKING 1, GROWING 1, WALKING 0, GROWING 0.  A SEEK
3637                                    ;IS ISSUED BEFORE EACH READ TO INSURE WE ARE ON THE PROPER
3638                                    ;TRACK.  ONCE WE ARE ON THE RIGHT TRACKWE LOAD THE RLDA
3639                                    ;AND ISSUE THE READ.  UPON COMPLETION WE WILL CHECK FOR ERRORS
3640                                    ;WE THEN LOAD THE COMPLEMENT PATTERN INTO THE RLDA
3641                                    ;EXPECTING A HEADER NOT FOUND TO SET
3642 022326                             STARS
                                        ;;*****************************************************************
3643
3644 022326  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3645 022332                                     CKERFG                  ;HEADS GO HOME OKAY
     022340  104432                             TRAP    C$EXIT
     022342  000574                             .WORD   L10056-.
3646
3647 022344                                     BGNSEG                          ;**START OF SEGMENT**
     022344  104404                             TRAP    C$BSEG
3648
3649                                    ;       SETPRI  #PRI07          ;PRIORITY TO 7                  ;JSD REV A
3650 022346                                     SETPRI  #PRI06          ;PRIORITY TO 6                  ;JSD REV A
     022346  012700  000300                     MOV     #PRI06,R0
     022352  104441                             TRAP    C$SPRI
3651 022354  022737  000001  002232            CMP     #1,T.DRIVE      ;CHECK TYPE OF DRIVE (RL01 OR RL02)
3652 022362  001003                             BNE     22$             ;RL02? THEN BRANCH
3653 022364  012703  002670                     MOV     #HDRTAB,R3      ;MOV ADDRESS OF BEG PATTERN TO R3
3654 022370  000402                             BR      33$             ; THEN BRANCH
3655 022372  012703  003050            22$:    MOV     #HTAB,R3        ;MOV ADDRESS OF BEG PATTERN TO R3
3656 022376                             33$:    BGNSEG                          ;START OF SEGMENT
     022376  104404                             TRAP    C$BSEG
3657 022400                             1$:
3658 022400  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3659 022404  000010                             RDHDR                   ;READ HEADER
3660 022406  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLRE READY
3661 022412                                     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022412  104410                             TRAP    C$ESCAPE
     022414  000516                             .WORD   10001$-.
3662
3663 022416  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3664 022422                                     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022422  104410                             TRAP    C$ESCAPE
     022424  000506                             .WORD   10001$-.
3665 022426  013737  002346  002274            MOV     E.MP,TMP1       ;READ AND SAVE HEADER
3666
3667 022434  042737  000177  002274            BIC     #177,TMP1       ;CLEAR OUT SECTOR AND H.S.
3668 022442  012777  000001  157710            MOV     #1,@RLDA        ;SETUP MARKER FOR SEEK
3669 022450  011337  002276                     MOV     (R3),TMP2       ;GET HEADER PATTERN
3670 022454  042737  000177  002276            BIC     #177,TMP2       ;CLEAR OUT SECTOR AND H.S.
3671 022462  163737  002274  002276            SUB     TMP1,TMP2       ;CALCULATE DIFFERENCE TO SEEK
3672 022470  103404                             BCS     2$              ;BRANCH FOR SEEK OUT
3673 022472  052777  000004  157660            BIS     #SIGN,@RLDA     ;SEEK TOWARDS SPINDLE
3674 022500  000402                             BR      3$              ;GO PUT IN DIFFERENCE WORD
3675 022502  005437  002276            2$:     NEG     TMP2            ;WE HAVE TO NEGATE DIFFERENCE
3676 022506  053777  002276  157644    3$:     BIS     TMP2,@RLDA      ;SET IN DIFFERENCE WORD
3677 022514  032713  000100                     BIT     #RHHS,(R3)      ;DO WE WANT HEAD SELECT AS 0?
3678 022520  001403                             BEQ     4$              ;YES, SKIP OVER SETTING H.S.
3679 022522  052777  000020  157630            BIS     #DAHS,@RLDA     ;SET HEAD SELECT TO ONE
3680 022530  004537  015054            4$:     JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
```

TEST 20 - CHECK HEADER COMPARE LOGIC

```
3681 022534  000006                        SEEK                    ;SEEK
3682
3683 022536  004537  015700                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3684 022542                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022542  104410                        TRAP    C$ESCAPE
     022544  000366                        .WORD   10001$-.
3685
3686 022546  004537  014612                JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3687 022552                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022552  104410                        TRAP    C$ESCAPE
     022554  000356                        .WORD   10001$-.
3688
3689 022556  004537  015634                JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
3690 022562                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022562  104410                        TRAP    C$ESCAPE
     022564  000346                        .WORD   10001$-.
3691 022566  004537  015054                JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3692 022572  000010                        RDHDR                   ;READ HEADER (VERIFY SEEK)
3693 022574  004537  015700                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3694 022600                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022600  104410                        TRAP    C$ESCAPE
     022602  000330                        .WORD   10001$-.
3695
3696 022604  004537  014612                JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3697 022610                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022610  104410                        TRAP    C$ESCAPE
     022612  000320                        .WORD   10001$-.
3698
3699 022614  013737  002346  002302        MOV     E.MP,BDDAT      ;READ HEADER
3700 022622  043737  002262  002302        BIC     SECMSK,BDDAT    ;SAVE CYLINDER FOR COMPARE
3701 022630  011337  002300                MOV     (R3),GDDAT      ;GET EXPECTED HEADER
3702 022634  043737  002262  002300        BIC     SECMSK,GDDAT    ;SAVE CYLINDER FOR COMPARE
3703 022642  023737  002300  002302        CMP     GDDAT,BDDAT     ;SEEK END UP OKAY
3704 022650  001404                        BEQ     5$              ;YES, CONTINUE
3705
3706 022652                                ERRDF   27.,EM11,ERR4   ;SEEK INCORRECT
     022652  104455                        TRAP    C$ERDF
     022654  000033                        .WORD   27
     022656  005124                        .WORD   EM11
     022660  007654                        .WORD   ERR4
3707
3708 022662                          5$:   ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022662  104410                        TRAP    C$ESCAPE
     022664  000246                        .WORD   10001$-.
3709
3710 022666  011377  157466                MOV     (R3),@RLDA      ;SET UP DISK ADDRESS
3711 022672  042777  000077  157460        BIC     #77,@RLDA
3712 022700  012777  177777  157454        MOV     #-1,@RLMP       ;WORD COUNT
3713 022706  012777  003426  157442        MOV     #BUF,@RLBA      ;BUS ADDRESS
3714
3715 022714  004537  015054                JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3716 022720  000014                        READ                    ;READ
3717 022722  004537  015700                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3718 022726                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022726  104410                        TRAP    C$ESCAPE
     022730  000202                        .WORD   10001$-.
3719
```

TEST 20 - CHECK HEADER COMPARE LOGIC

```
3720 022732  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3721 022736                                     ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     022736  104410                             TRAP    C$ESCAPE
     022740  000172                             .WORD   10001$-.
3722
3723 022742  011377  157412                    MOV     (R3),@RLDA              ;SET UP DISK ADDRESS AS
3724 022746  005177  157406                    COM     @RLDA                   ;COMPLIMENT TO CAUSE HDR NT FND
3725 022752  012777  177777  157402            MOV     #-1,@RLMP               ;WORD COUNT
3726 022760  012777  003426  157370            MOV     #BUF,@RLBA              ;BUS ADDRESS
3727
3728 022766  004537  015054                    JSR     R5,LDFUNC                       ;LOAD THE FUNCTION IN NEXT WORD
3729 022772  000014                            READ                            ;READ
3730 022774  004537  015700                    JSR     R5,WTCRDY               ;WAIT FOR CONTROLLER READY
3731 023000                                     ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     023000  104410                             TRAP    C$ESCAPE
     023002  000130                             .WORD   10001$-.
3732 023004  013737  002340  002272            MOV     E.CS,TMP0               ;GET CS
3733 023012  042737  001777  002272            BIC     #1777,TMP0              ;SAVE ERROR BITS
3734 023020  022737  112000  002272            CMP     #BIT15!BIT12!BIT10,TMP0  ;DID HEADER NOT FOUND SET
3735 023026  001402                            BEQ     8$                      ;YES, CONTINUE
3736 023030  004537  014612                    JSR     R5,CHERR
3737 023034                             8$:     CKLOOP
     023034  104406                             TRAP    C$CLP1
3738
3739 023036  022737  112000  002272            CMP     #BIT15!BIT12!BIT10,TMP0
3740 023044  001413                            BEQ     6$
3741
3742 023046  011337  002300                    MOV     (R3),GDDAT              ;SET UP DATA FOR ERROR
3743 023052  013737  002300  002302            MOV     GDDAT,BDDAT             ;PRINT OUT
3744 023060  005137  002302                    COM     BDDAT
3745
3746 023064                                     ERRDF   28.,EM12,ERR4           ;HDR NOT FOUND WOULD NOT SET
     023064  104455                             TRAP    C$ERDF
     023066  000034                             .WORD   28
     023070  005144                             .WORD   EM12
     023072  007654                             .WORD   ERR4
3747
3748 023074                             6$:     ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     023074  104410                             TRAP    C$ESCAPE
     023076  000034                             .WORD   10001$-.
3749
3750 023100  005723                            TST     (R3)+                   ;GET NEXT PATTERN
3751 023102  022737  000001  002232            CMP     #1,T.DRIVE              ;TYPE OF DRIVE RL01 OR RL02
3752 023110  001003                            BNE     60$                     ;RL02 ? THEN BRANCH
3753 023112  020327  003046                    CMP     R3,#HDREND              ;CMP IT WITH #HDREND
3754 023116  000402                            BR      77$                     ;THEN BRANCH
3755 023120  020327  003234             60$:   CMP     R3,#END                 ;CMP IT WITH #END
3756 023124  001402                     77$:   BEQ     7$                      ;YES,EXIT TEST
3757 023126  000137  022400                    JMP     1$                      ;NO, GO BACK
3758
3759 023132                             7$:
3760 023132                                     ENDSEG                                  ;**END OF SEGMENT**
     023132                             10001$:
     023132  104405                             TRAP    C$ESEG
3761
3762 023134                                     ENDSEG                                  ;**END OF SEGMENT**
     023134                             10000$:
```

```
**TEST 20** - CHECK HEADER COMPARE LOGIC

      023134  104405                          TRAP    C$ESEG
 3763 023136                          ENDTST                                  ;**END OF TEST**
      023136                          L10056:
      023136  104401                          TRAP    C$ETST
 3764
 3765                                 .SBTTL  **TEST 21** - CHECK MULTIPLE SECTORS ON READ
 3766
 3767 023140                          BGNTST                                  ;**START OF TEST**
 3768
 3769 023140                          STARS
                                      ;;**********************************************************************
 3770                                 ;VERIFY THAT MULTIPLE SECTORS CAN BE READ, WE WILL CHECK
 3771                                 ;THAT THE RLDA INCREMENTS PROPERLY.
 3772 023140                          STARS
                                      ;;**********************************************************************
 3773
 3774 023140  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 3775 023144                                  CKERFG                  ;HEADS GO HOME OKAY
      023152  104432                          TRAP    C$EXIT
      023154  000156                          .WORD   L10057-.
 3776
 3777 023156  005037  002272                  CLR     TMP0            ;CLEAR LOCATIONS
 3778 023162  005037  002274                  CLR     TMP1
 3779
 3780 023166                                  BGNSEG                          ;**START OF SEGMENT**
      023166  104404                          TRAP    C$BSEG
 3781
 3782 023170                          1$:
 3783 023170  013737  002274  002300          MOV     TMP1,GDDAT      ;GET CYLINDER
 3784 023176  053737  002272  002300          BIS     TMP0,GDDAT      ;GET SECTOR
 3785 023204  013777  002300  157146          MOV     GDDAT,@RLDA     ;SET DISK ADDRESS-SECTOR 0
 3786 023212  062737  000002  002300          ADD     #2,GDDAT        ;SET EXPECTED + 2
 3787 023220  012777  003426  157130          MOV     #BUF,@RLBA      ;SET BUS ADDRESS
 3788 023226  012777  177577  157126          MOV     #-129.,@RLMP    ;WORD COUNT-SECTOR+1 WORD
 3789
 3790 023234  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 3791 023240  000014                          READ                    ;READ
 3792 023242  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY?
 3793 023246                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      023246  104410                          TRAP    C$ESCAPE
      023250  000060                          .WORD   10000$-.
 3794
 3795 023252  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 3796 023256                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      023256  104410                          TRAP    C$ESCAPE
      023260  000050                          .WORD   10000$-.
 3797
 3798 023262  013737  002344  002302          MOV     E.DA,BDDAT      ;READ DISK ADDRESS
 3799 023270  023737  002302  002300          CMP     BDDAT,GDDAT     ;IS DISK ADDRESS CORRECT
 3800 023276  001404                          BEQ     2$              ;YES, BRANCH  NO, REPORT ERROR
 3801
 3802 023300                                  ERRDF   29.,EM14,ERR4   ;DA DID NOT INCREMENT
      023300  104455                          TRAP    C$ERDF
      023302  000035                          .WORD   29
      023304  005224                          .WORD   EM14
      023306  007654                          .WORD   ERR4
 3803
```

TEST 21 - CHECK MULTIPLE SECTORS ON READ

```
3804 023310                          2$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     023310  104410                          TRAP    C$ESCAPE
     023312  000016                          .WORD   10000$-.
3805
3806 023314  005237  002272                 INC     TMP0            ;NEXT SECTOR?
3807 023320  022737  000046  002272         CMP     #46,TMP0        ;DONE?
3808 023326  001320                         BNE     1$              ;NO, GO BACK
3809
3810 023330                                 ENDSEG                          ;**END OF SEGMENT**
     023330                          10000$:
     023330  104405                          TRAP    C$ESEG
3811 023332                          ENDTST                          ;**END OF TEST**
     023332                          L10057:
     023332  104401                          TRAP    C$ETST
3812 023334                          STARS
                                     ;;**************************************************************************
3813                                 ;CHECK THAT WE CAN FORCE A HEADER NOT FOUND AT THE
3814                                 ;END OF A TRACK DOING A MULTIPLE SECTOR READ.  WE
3815                                 ;SET UP TO READ TWO SECTORS STARTING AT SECTOR 39
3816                                 ;WE SHOULD TRANSFER 128 WORDS THEN ABORT WITH A
3817                                 ;HEADER NOT FOUND FOR SECTOR 40
3818 023334                          STARS
                                     ;;**************************************************************************
3819
3820                                 .SBTTL  **TEST 22** - FORCE HDR NT FND AT END OF TRACK
3821
3822 023334                          BGNTST                          ;**START OF TEST**
3823
3824 023334  004737  015764                 JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3825 023340                                 CKERFG                  ;HEADS GO HOME OKAY
     023346  104432                         TRAP    C$EXIT
     023350  000126                         .WORD   L10060-.
3826
3827 023352                                 BGNSEG                          ;**START OF SEGMENT**
     023352  104404                         TRAP    C$BSEG
3828
3829 023354  012737  000047  002300         MOV     #39.,GDDAT      ;CREATE LAST SECTOR
3830 023362  013777  002300  156770         MOV     GDDAT,@RLDA     ;LOAD DISK ADDRESS
3831 023370  012777  177577  156764         MOV     #-129.,@RLMP    ;WORD COUNT
3832 023376  012777  003426  156752         MOV     #BUF,@RLBA      ;BUS ADDRESS
3833 023404  004537  015054                 JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3834 023410  000014                         READ                    ;READ
3835 023412  004537  015700                 JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3836 023416                                 ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     023416  104410                         TRAP    C$ESCAPE
     023420  000054                         .WORD   10000$-.
3837
3838 023422  013737  002340  002302         MOV     E.CS,BDDAT      ;READ CS
3839 023430  042737  001777  002302         BIC     #1777,BDDAT     ;SAVE ERROR BITS
3840 023436  022737  112000  002302         CMP     #112000,BDDAT   ;HDR NOT FOUND SET?
3841 023444  001402                         BEQ     4$              ;YES, CONTINUE
3842 023446  004537  014612                 JSR     R5,CHERR
3843 023452                          4$:     CKLOOP
     023452  104406                         TRAP    C$CLP1
3844
3845 023454  022737  112000  002302         CMP     #112000,BDDAT
3846 023462  001404                         BEQ     1$
```

TEST 22 - FORCE HDR NT FND AT END OF TRACK

```
3847
3848 023464                                ERRDF   30.,EM23,ERRO   ;HEADER NOT FOUND DID NOT SET
     023464  104455                        TRAP    C$ERDF
     023466  000036                        .WORD   30
     023470  005540                        .WORD   EM23
     023472  007510                        .WORD   ERRO
3849
3850 023474                           1$:
3851
3852 023474                                ENDSEG                          ;##END OF SEGMENT##
     023474                           10000$:
     023474  104405                        TRAP    C$ESEG
3853 023476                           ENDTST                               ;**END OF TEST**
     023476                           L10060:
     023476  104401                        TRAP    C$ETST
3854
3855                                  .SBTTL  **TEST 23** - FORCE NON-EXISTENT MEMORY ERROR
3856
3857 023500                           BGNTST                               ;**START OF TEST**
3858
3859 023500                           STARS
                                      ;;*****************************************************************
3860                                  ;CHECK FOR RLV-11
3861                                  ; &
3862                                  ;SIZE IF MEMORY >= 124K - IF TRUE DO NOT PERFORM TESTS 23 & 24
3863 023500                           STARS
                                      ;;*****************************************************************
3864
3865 023500  005037  002662                CLR     NOTST           ;INIT ABORT TEST
3866 023504  005737  002420                TST     T.CNTLR         ;RLV11?
3867 023510  001013                        BNE     4$              ;BRANCH - IF NO
3868 023512  013700  002120                MOV     L$HIMEM,R0      ;GET HIGHEST OCTAL MEMORY ADDRESS IN PAR FORMAT
3869 023516  006200                        ASR     R0              ;DIVIDE BY
3870 023520  006200                        ASR     R0              ;32(10),40(8)
3871 023522  006200                        ASR     R0              ;TO CONVERT TO
3872 023524  006200                        ASR     R0              ;1K(10)
3873 023526  006200                        ASR     R0              ;BLOCKS
3874 023530  005200                        INC     R0              ;TO INCLUDE LOCATION ZERO
3875 023532  022700  000174                CMP     #124.,R0        ;MEMORY >= 124K.?
3876 023536  003447                        BLE     5$              ;BRANCH - IF YES
3877
3878 023540                           STARS
                                      ;;*****************************************************************
3879                                  ;FORCE A NON-EXISTENT MEMORY ERROR,
3880                                  ;WE SET THE RLBA TO EQUAL THE
3881                                  ;LAST ADDRESS IN MEMORY AND ISSUE A READ.  THE
3882                                  ;READ SHOULD ABORT AFTER ONE WORD TRANSFERRED
3883 023540                           STARS
                                      ;;*****************************************************************
3884
3885 023540  004737  015764           4$:     JSR     PC,HDHOME       ;HEADS OVER TRACK 0
3886 023544                                   CKERFG                  ;HEADS GO HOME OKAY
     023552  104432                        TRAP    C$EXIT
     023554  000106                        .WORD   L10061-.
3887
3888 023556                                BGNSEG                          ;##START OF SEGMENT##
     023556  104404                        TRAP    C$BSEG
```

```
**TEST 23** - FORCE NON-EXISTENT MEMORY ERROR

 3889
 3890 023560  012777  160000  156570          MOV     #160000,@RLBA   ;LEAD BA
 3891 023566  012737  000060  002374          MOV     #BA16!BA17,XMEM ;SET EA BIT
 3892 023574  005077  156560                  CLR     @RLDA           ;LOAD DISK AVAILABLE
 3893 023600  012777  177600  156554          MOV     #-128.,@RLMP    ;WORD COUNT
 3894 023606  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 3895 023612  000014                          READ                    ;READ
 3896 023614  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER
 3897 023620                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      023620  104410                          TRAP    C$ESCAPE
      023622  000026                          .WORD   10000$-.
 3898 023624  032737  020000  002340          BIT     #NXM,E.CS       ;DID NXM SET?
 3899 023632  001004                          BNE     3$              ;YES, CONTINUE
 3900 023634                                  ERRDF   31.,EM24,ERR0   ;NXM DID NOT SET
      023634  104455                          TRAP    C$ERDF
      023636  000037                          .WORD   31
      023640  005616                          .WORD   EM24
      023642  007510                          .WORD   ERR0
 3901 023644                          3$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      023644  104410                          TRAP    C$ESCAPE
      023646  000002                          .WORD   10000$-.
 3902 023650                                  ENDSEG                  ;**END OF SEGMENT**
      023650                          10000$:
      023650  104405                          TRAP    C$ESEG
 3903 023652                                  EXIT    TST
      023652  104432                          TRAP    C$EXIT
      023654  000006                          .WORD   L10061-.
 3904 023656  005237  002662          5$:     INC     NOTST           ;ABORT TEST 24
 3905
 3906 023662                          ENDTST                          ;**END OF TEST**
      023662                          L10061:
      023662  104401                          TRAP    C$ETST
 3907
 3908                                 .SBTTL  **TEST 24** - FORCE NON-EXISTENT MEMORY ERROR INTERRUPT
 3909
 3910 023664                          BGNTST                          ;**START OF TEST**
 3911 023664                          STARS
                                      ;;*************************************************************************
 3912                                 ;CHECK THAT WE CAN FORCE AN INTERRUPT WITH A
 3913                                 ;NON-EXISTENT MEMORY ERROR.
 3914 023664                          STARS
                                      ;;*************************************************************************
 3915
 3916 023664  005737  002662                  TST     NOTST           ;RLV-11 & MEMORY SIZE >= 124K.?
 3917 023670  001066                          BNE     1$              ;BRANCH - IF YES
 3918 023672  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 3919 023676                                  CKERFG                  ;HEADS GO HOME OKAY
      023704  104432                          TRAP    C$EXIT
      023706  000140                          .WORD   L10062-.
 3920
 3921 023710                                  BGNSEG                          ;**START OF SEGMENT**
      023710  104404                          TRAP    C$BSEG
 3922
 3923 023712  005037  002256                  CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE FLAG
 3924 023716                                  SETPRI  #PRI00
      023716  012700  000000                  MOV     #PRI00,R0
      023722  104441                          TRAP    C$SPRI
```

```
**TEST 24** - FORCE NON-EXISTENT MEMORY ERROR INTERRUPT

3925 023724  012777  160000  156424          MOV     #160000,@RLBA   ;PRELOAD BA
3926 023732  012737  000060  002374          MOV     #BA16!BA17,XMEM ;SET EA BITS
3927 023740  005077  156414                  CLR     @RLDA           ;LOAD DA
3928 023744  012777  177777  156410          MOV     #-1,@RLMP       ;WORD COUNT
3929 023752  004537  015054                  JSR     R5,LDFUNC       ;LOAD THE FUNCTION IN NEXT WORD
3930 023756  000114                          READ!INTEN              ;READ
3931 023760  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER
3932                                 ;       SETPRI  #PRI07          ;PRIORITY TO 7                  ;JSD REV A
3933 023764                                  SETPRI  #PRI06          ;PRIORITY TO 6                  ;JSD REV A
     023764  012700  000300                  MOV     #PRI06,R0
     023770  104441                          TRAP    C$SPRI
3934 023772                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     023772  104410                          TRAP    C$ESCAPE
     023774  000050                          .WORD   10000$-.
3935 023776  005737  002256                  TST     INTFLG          ;INTERRUPT OCCUR?
3936 024002  001004                          BNE     4$              ;YES OKAY
3937 024004                                  ERRDF   32.,EM44,ERR0   ;NO INTERRUPT W/NXM
     024004  104455                          TRAP    C$ERDF
     024006  000040                          .WORD   32
     024010  006522                          .WORD   EM44
     024012  007510                          .WORD   ERR0
3938 024014                          4$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024014  104410                          TRAP    C$ESCAPE
     024016  000026                          .WORD   10000$-.
3939 024020  032737  020000  002340          BIT     #NXM,E.CS       ;DID NXM SET?
3940 024026  001004                          BNE     3$              ;YES, CONTINUE
3941 024030                                  ERRDF   33.,EM24,ERR0   ;NO NXM
     024030  104455                          TRAP    C$ERDF
     024032  000041                          .WORD   33
     024034  005616                          .WORD   EM24
     024036  007510                          .WORD   ERR0
3942 024040                          3$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024040  104410                          TRAP    C$ESCAPE
     024042  000002                          .WORD   10000$-.
3943 024044                                  ENDSEG                  ;**END OF SEGMENT**
     024044                          10000$:
     024044  104405                          TRAP    C$ESEG
3944 024046                          1$:
3945
3946 024046                          ENDTST                          ;**END OF TEST**
     024046                          L10062:
     024046  104401                          TRAP    C$ETST
3947
3948                                 .SBTTL  **TEST 25** - CHECK READ WRITE LOOP
3949
3950 024050                          BGNTST                                  ;**START OF TEST**
3951
3952 024050                          STARS
                                     ;;**********************************************************************
3953                                 ;VERIFY THAT THE WRITE ACTUALLY WRITES.  AT THIS
3954                                 ;TIME WE KNOW THAT THE WRITE FUNCTION GOES THRU
3955                                 ;THE MOTIONS BUT WE DON'T KNOW THAT THE DATA
3956                                 ;ACTUALLY GETS RECORDED ON THE PLATTER.
3957 024050                          STARS
                                     ;;**********************************************************************
3958
3959 024050  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
```

TEST 25 - CHECK READ WRITE LOOP

```
3960 024054                                    CKERFG                  ;HEADS GO HOME OKAY
     024062  104432                            TRAP    C$EXIT
     024064  000362                            .WORD   L10063-.
3961
3962 024066                                    BGNSEG                          ;%%START OF SEGMENT%%
     024066  104404                            TRAP    C$BSEG
3963
3964 024070  012700  003426                    MOV     #BUF,R0         ;SET UP WRITE BUFFER
3965 024074  012701  000200                    MOV     #128.,R1        ;128 WORDS/ONE SECTOR
3966 024100  012720  125252            3$:     MOV     #125252,(R0)+   ;WRITE PATTERN TO BUFFER
3967 024104  005301                            DEC     R1              ;DONE?
3968 024106  001374                            BNE     3$              ;NO, BRANCH BACK
3969 024110  005077  156244                    CLR     @RLDA           ;DISK ADDRESS
3970 024114  012777  177600  156240            MOV     #-128.,@RLMP    ;WORD COUNT
3971 024122  012777  003426  156226            MOV     #BUF,@RLBA      ;BUS ADDRESS
3972 024130  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3973 024134  000012                            WRITE                   ;WRITE THE PATTERN
3974 024136  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3975 024142                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024142  104410                            TRAP    C$ESCAPE
     024144  000300                            .WORD   10000$-.
3976
3977 024146  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3978 024152                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024152  104410                            TRAP    C$ESCAPE
     024154  000270                            .WORD   10000$-.
3979 024156                                    BGNSEG                          ;%%START OF SEGMENT%%
     024156  104404                            TRAP    C$BSEG
3980 024160  012700  003426                    MOV     #BUF,R0         ;CLEAR OUT BUFFER BEFORE
3981 024164  012701  000200                    MOV     #128.,R1        ;READING
3982 024170  005020            4$:     CLR     (R0)+           ;CLEAR BUFFER
3983 024172  005301                            DEC     R1              ;DONE?
3984 024174  001375                            BNE     4$              ;NO, BRANCH BACK
3985 024176  005077  156156                    CLR     @RLDA           ;LOAD DISK ADDRESS
3986 024202  012777  177600  156152            MOV     #-128.,@RLMP    ;WORD COUNT/ONE SECTION
3987 024210  012777  003426  156140            MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
3988 024216  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
3989 024222  000014                            READ                    ;GO READ
3990 024224  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
3991 024230                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024230  104410                            TRAP    C$ESCAPE
     024232  000210                            .WORD   10001$-.
3992
3993 024234  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
3994 024240  005737  002236                    TST     T.CRC           ;WAS ERROR A DCK??
3995 024244  001003                            BNE     8$              ;YES,SEE IF WE A DUMP
3996 024246                            10$:    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024246  104410                            TRAP    C$ESCAPE
     024250  000172                            .WORD   10001$-.
3997 024252  000404                            BR      99$             ;SKIP AROUND
3998 024254  005737  012440            8$:     TST     T.DMP           ;DO WE STILL WANT TO CHECK IT
3999 024260  001772                            BEQ     10$             ;NO
4000 024262                                    CKLOOP                  ;YES, CHECK FOR LOOP FIRST
     024262  104406                            TRAP    C$CLP1
4001
4002 024264  005037  002242            99$:    CLR     CDCNT           ;CLEAR NUMBER WE'RE TO PRINT
4003 024270  005037  002234                    CLR     CHECK           ;ALLOW HEADER ON FIRST PRINT
```

```
**TEST 25** - CHECK READ WRITE LOOP

 4004 024274  012702  003426                    MOV     #BUF,R2          ;COMPARE BUFFER TO CHECK WRITE
 4005 024300  012701  000200                    MOV     #128.,R1         ;128 WORDS
 4006 024304  012737  125252  002300            MOV     #125252,GDDAT    ;SET UP EXPECTED
 4007 024312  011237  002302              5$:   MOV     (R2),BDDAT       ;GET DATA
 4008 024316  023737  002300  002302            CMP     GDDAT,BDDAT      ;IS DATA OKAY
 4009 024324  001442                            BEQ     6$               ;YES, CONTINUE
 4010 024326  010237  002274                    MOV     R2,TMP1          ;LOAD BAD MEM LOCATION
 4011 024332  023737  002242  012442            CMP     CDCNT,T.LMT              ;CHECKED ENOUGH??
 4012 024340  001002                            BNE     333$             ;NO
 4013 024342                                    ESCAPE  SEG              ;CHECK FOR FL:LOE, ELSE EXIT SEG
      024342  104410                            TRAP    C$ESCAPE
      024344  000076                            .WORD   10001$-.
 4014 024346  005237  002242            333$:  INC     CDCNT            ;ACCOUNT FOR IT
 4015
 4016 024352  005737  002234                    TST     CHECK            ;HEADER OR JUST DATA
 4017 024356  001007                            BNE     9$               ;JUST DATA
 4018 024360                                    ERRDF   34.,EM25,ERR8    ;BAD DATA
      024360  104455                            TRAP    C$ERDF
      024362  000042                            .WORD   34
      024364  005656                            .WORD   EM25
      024366  010030                            .WORD   ERR8
 4019 024370  005237  002234                    INC     CHECK            ;ACCOUNT FOR PRINT OF HEADER
 4020 024374  000416                            BR      6$
 4021
 4022 024376                              9$:   PRINTB  #FRMT6,TMP1,GDDAT,BDDAT
      024376  013746  002302                    MOV     BDDAT,-(SP)
      024402  013746  002300                    MOV     GDDAT,-(SP)
      024406  013746  002274                    MOV     TMP1,-(SP)
      024412  012746  011277                    MOV     #FRMT6,-(SP)
      024416  012746  000004                    MOV     #4,-(SP)
      024422  010600                            MOV     SP,R0
      024424  104414                            TRAP    C$PNTB
      024426  062706  000012                    ADD     #12,SP
 4023
 4024 024432                              6$:   CKLOOP
      024432  104406                            TRAP    C$CLP1
 4025 024434  005722                      7$:   TST     (R2)+            ;BUMP BUFFER POINTER
 4026 024436  005301                            DEC     R1               ;DONE?
 4027 024440  001324                            BNE     5$               ;NO, GO BACK
 4028 024442                                    ENDSEG                           ;**END OF SEGMENT**
      024442                            10001$:
      024442  104405                            TRAP    C$ESEG
 4029 024444                                    ENDSEG                           ;**END OF SEGMENT**
      024444                            10000$:
      024444  104405                            TRAP    C$ESEG
 4030 024446                            ENDTST                                  ;**END OF TEST**
      024446                            L10063:
      024446  104401                            TRAP    C$ETST
 4031
 4032                                   .SBTTL  **TEST 26** - CHECK SILO LINES
 4033
 4034 024450                            BGNTST                                  ;**START OF TEST**
 4035
 4036 024450                            STARS
                                        ;;**********************************************************************
 4037                                   ;TEST THAT LINES IN / TO SILO ARE GOOD, THAT IS THAT EACH LINE IS
 4038                                   ;GOOD AND CAN BE AT EITHER A 1 OR A 0 STATE INDEPENDENTLY OF EACH
```

TEST 26 - CHECK SILO LINES

```
4039                                    ;OTHER BIT POSITION THIS IS DONE BY WRITING PATTERNS OF FLOATING 1,
4040                                    ;FLOATING 0, WALKING 0, WALKING 1
4041 024450                             STARS
                                        ;;*****************************************************************
4042
4043 024450  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
4044 024454                                    CKERFG                  ;HEADS GO HOME OKAY
     024462  104432                            TRAP    C$EXIT
     024464  000404                            .WORD   L10064-.
4045
4046 024466  012703  003236                    MOV     #DATPAT,R3
4047
4048 024472                                    BGNSEG                          ;**START OF SEGMENT**
     024472  104404                            TRAP    C$BSEG
4049 024474  012700  003426            6$:     MOV     #BUF,R0         ;WRITE PATTERN INTO MEMORY
4050 024500  012701  000200                    MOV     #128.,R1        ;128 WORDS
4051 024504  011320            2$:     MOV     (R3),(R0)+      ;WRITE THE PATTERN
4052 024506  005301                    DEC     R1              ;DONE?
4053 024510  001375                    BNE     2$              ;NO GO BACK
4054
4055 024512  012777  003426  155636            MOV     #BUF,@RLBA      ;SETUP TO WRITE PATTERN ONTO DISK
4056 024520  005077  155634                    CLR     @RLDA           ;LOAD DA
4057 024524  012777  177600  155630            MOV     #-128.,@RLMP    ;WORD COUNT
4058 024532  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4059 024536  000012                            WRITE
4060 024540  004537  015700                    JSR     R5,WTCRDY
4061 024544                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024544  104410                            TRAP    C$ESCAPE
     024546  000320                            .WORD   10000$-.
4062 024550  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4063 024554                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024554  104410                            TRAP    C$ESCAPE
     024556  000310                            .WORD   10000$-.
4064 024560                                    BGNSEG                          ;**START OF SEGMENT**
     024560  104404                            TRAP    C$BSEG
4065 024562  012700  003426                    MOV     #BUF,R0         ;CLEAR MEMORY BEFORE READING IT BACK
4066 024566  012701  000200                    MOV     #128.,R1        ;128 WORDS
4067 024572  005020            3$:     CLR     (R0)+           ;CLEAR
4068 024574  005301                    DEC     R1              ;EONE
4069 024576  001375                    BNE     3$              ;NO
4070
4071 024600  012777  003426  155550            MOV     #BUF,@RLBA      ;SETUP TO READ IT BACK
4072 024606  012777  177600  155546            MOV     #-128.,@RLMP    ;128 WORDS
4073 024614  005077  155540                    CLR     @RLDA           ;SECTOR ZERO
4074 024620  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4075 024624  000014                            READ
4076 024626  004537  015700                    JSR     R5,WTCRDY
4077 024632                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024632  104410                            TRAP    C$ESCAPE
     024634  000224                            .WORD   10001$-.
4078 024636  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4079 024642  005737  002236                    TST     T.CRC           ;WAS ERROR A DCK??
4080 024646  001003                    BNE     8$              ;YES,SEE IF WE A DUMP
4081 024650            10$:    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024650  104410                            TRAP    C$ESCAPE
     024652  000206                            .WORD   10001$-.
4082 024654  000404                    BR      99$             ;SKIP AROUND
```

TEST 26 - CHECK SILO LINES

```
4083 024656  005737  012440          8$:    TST     T.DMP           ;DO WE STILL WANT TO CHECK IT
4084 024662  001772                         BEQ     10$             ;NO
4085 024664                                  CKLOOP                  ;YES, CHECK FOR LOOP FIRST
     024664  104406                         TRAP    C$CLP1
4086
4087 024666  005037  002242          99$:   CLR     CDCNT           ;CLEAR NUMBER WE'RE TO PRINT
4088 024672  005037  002234                 CLR     CHECK           ;ALLOW HEADER ON FIRST PRINT
4089 024676  011337  002300                 MOV     (R3),GDDAT      ;COMPARE WHAT WE READ BACK
4090 024702  012737  003426  002276         MOV     #BUF,TMP2       ;BUFFER START
4091 024710  012737  000001  002274         MOV     #1,TMP1         ;START WITH FIRST
4092
4093 024716  017737  155354  002302  5$:    MOV     @TMP2,BDDAT     ;GET DATA
4094 024724  023737  002300  002302         CMP     GDDAT,BDDAT     ;GOOD?
4095 024732  001440                         BEQ     4$              ;YES, BRANCH
4096
4097 024734  023737  002242  012442         CMP     CDCNT,T.LMT             ;CHECKED ENOUGH??
4098 024742  001002                         BNE     333$            ;NO
4099 024744                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     024744  104410                         TRAP    C$ESCAPE
     024746  000112                         .WORD   10001$-.
4100 024750  005237  002242          333$:  INC     CDCNT           ;ACCOUNT FOR IT
4101
4102 024754  005737  002234                 TST     CHECK           ;HEADER OR JUST DATA
4103 024760  001007                         BNE     9$              ;JUST DATA
4104 024762                                  ERRDF   35.,EM45,ERR10  ;BAD DATA BACK
     024762  104455                         TRAP    C$ERDF
     024764  000043                         .WORD   35
     024766  006554                         .WORD   EM45
     024770  010146                         .WORD   ERR10
4105
4106 024772  005237  002234                 INC     CHECK           ;ACCOUNT FOR PRINT OF HEADER
4107 024776  000416                         BR      4$
4108
4109 025000                          9$:    PRINTB  #FRMT7,TMP1,GDDAT,BDDAT
     025000  013746  002302                 MOV     BDDAT,-(SP)
     025004  013746  002300                 MOV     GDDAT,-(SP)
     025010  013746  002274                 MOV     TMP1,-(SP)
     025014  012746  011354                 MOV     #FRMT7,-(SP)
     025020  012746  000004                 MOV     #4,-(SP)
     025024  010600                         MOV     SP,R0
     025026  104414                         TRAP    C$PNTB
     025030  062706  000012                 ADD     #12,SP
4110 025034                          4$:    CKLOOP
     025034  104406                         TRAP    C$CLP1
4111
4112 025036  062737  000002  002276         ADD     #2,TMP2         ;NEXT LOCATION
4113 025044  005237  002274                 INC     TMP1            ;NEXT WORD
4114 025050  023727  002274  000201         CMP     TMP1,#129.      ;DONE
4115 025056  001317                         BNE     5$              ;NO, GO BACK
4116
4117 025060                                  ENDSEG                          ;**END OF SEGMENT**
     025060                          10001$:
     025060  104405                         TRAP    C$ESEG
4118
4119 025062  005723                         TST     (R3)+           ;DONE ALL PATTERNS
4120 025064  001203                         BNE     6$              ;NO, GO BACK
4121
```

```
4122 025066                            ENDSEG                          ;##END OF SEGMENT##
     025066                      10000$:
     025066  104405                    TRAP    C$ESEG
4123 025070                      ENDTST                                ;**END OF TEST**
     025070                      L10064:
     025070  104401                    TRAP    C$ETST
4124
4125                                   .SBTTL  **TEST 27** - CHECK THROUGHPUT OF SILO
4126
4127 025072                            BGNTST                          ;**START OF TEST**
4128
4129 025072                            STARS
                                       ;;*************************************************************************
4130                                   ;TEST THAT THE SILO OPERATES CORRECTLY. WE WILL WRITE A PATTERN
4131                                   ;THAT CONTAINS A UNIQUE PATTERN IN EACH LOCATION. WE EXPECT IT
4132                                   ;BACK IN PROPER ORDER. WE DO A ONE SECTOR TRANSFER.
4133 025072                            STARS
                                       ;;*************************************************************************
4134
4135 025072  004737  015764                JSR     PC,HDHOME       ;HEADS OVER TRACK 0
4136 025076                                CKERFG                  ;HEADS GO HOME OKAY
     025104  104432                        TRAP    C$EXIT
     025106  000410                        .WORD   L10065-.
4137
4138 025110                                BGNSEG                          ;##START OF SEGMENT##
     025110  104404                        TRAP    C$BSEG
4139
4140 025112  012700  000001                MOV     #1,R0           ;INITIAL 1
4141 025116  012701  000200                MOV     #128.,R1        ;128 WORDS
4142 025122  012702  003426                MOV     #BUF,R2         ;BUFFER
4143 025126  010022                 2$:    MOV     R0,(R2)+        ;WRITE A WORD
4144 025130  005200                        INC     R0              ;NEXT PATTERN (1-128)
4145 025132  005301                        DEC     R1              ;DONE
4146 025134  001374                        BNE     2$              ;NO
4147
4148 025136  012777  003426  155212        MOV     #BUF,@RLBA      ;SETUP TO WRITE
4149 025144  012777  177600  155210        MOV     #-128.,@RLMP    ;128 WORDS
4150 025152  005077  155202                CLR     @RLDA           ;DISK ADDRESS 0
4151 025156  004537  015054                JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4152 025162  000012                        WRITE
4153 025164  004537  015700                JSR     R5,WTCRDY
4154 025170                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     025170  104410                        TRAP    C$ESCAPE
     025172  000322                        .WORD   10000$-.
4155
4156 025174  004537  014612                JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4157 025200                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     025200  104410                        TRAP    C$ESCAPE
     025202  000312                        .WORD   10000$-.
4158 025204                                BGNSEG                          ;##START OF SEGMENT##
     025204  104404                        TRAP    C$BSEG
4159 025206  012700  003426                MOV     #BUF,R0         ;CLEAR BUFFER
4160 025212  012701  000200                MOV     #128.,R1        ;128 IN LENGTH
4161 025216  005020                 3$:    CLR     (R0)+           ;CLEAR
4162 025220  005301                        DEC     R1              ;DOWN COUNT
4163 025222  001375                        BNE     3$              ;DONE?
4164
```

TEST 27 - CHECK THROUGHPUT OF SILO

```
4165 025224  012777  003426  155124          MOV     #BUF,@RLBA      ;BUS ADDRESS
4166 025232  012777  177600  155122          MOV     #-128.,@RLMP    ;WORD COUNT
4167 025240  005077  155114                  CLR     @RLDA           ;DISK ADDRESS
4168 025244  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4169 025250  000014                          READ
4170 025252  004537  015700                  JSR     R5,WTCRDY
4171 025256                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     025256  104410                          TRAP    C$ESCAPE
     025260  000232                          .WORD   10001$-.
4172
4173 025262  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4174 025266  005737  002236                  TST     T.CRC           ;WAS ERROR A DCK??
4175 025272  001003                          BNE     8$              ;YES,SEE IF WE A DUMP
4176 025274                          10$:    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     025274  104410                          TRAP    C$ESCAPE
     025276  000214                          .WORD   10001$-.
4177 025300  000404                          BR      99$             ;SKIP AROUND
4178 025302  005737  012440          8$:     TST     T.DMP           ;DO WE STILL WANT TO CHECK IT
4179 025306  001772                          BEQ     10$             ;NO
4180 025310                                  CKLOOP                  ;YES, CHECK FOR LOOP FIRST
     025310  104406                          TRAP    C$CLP1
4181
4182 025312  005037  002242          99$:    CLR     CDCNT           ;CLEAR NUMBER WE'RE TO PRINT
4183 025316  005037  002234                  CLR     CHECK           ;ALLOW HEADER ON FIRST PRINT
4184 025322  012737  000001  002300          MOV     #1,GDDAT        ;START GOOD AT 1
4185 025330  012737  003426  002276          MOV     #BUF,TMP2       ;START OF BUFFER
4186 025336  012737  000001  002274          MOV     #1,TMP1         ;FIRST WORD
4187
4188 025344  017737  154726  002302  4$:     MOV     @TMP2,BDDAT     ;GET WORD
4189 025352  023737  002302  002300          CMP     BDDAT,GDDAT     ;CORRECT?
4190 025360  001440                          BEQ     6$              ;YES
4191
4192 025362  023737  002242  012442          CMP     CDCNT,T.LMT             ;CHECKED ENOUGH??
4193 025370  001002                          BNE     333$            ;NO
4194 025372                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     025372  104410                          TRAP    C$ESCAPE
     025374  000116                          .WORD   10001$-.
4195 025376  005237  002242          333$:   INC     CDCNT           ;ACCOUNT FOR IT
4196
4197 025402  005737  002234                  TST     CHECK           ;HEADER OR JUST DATA
4198 025406  001007                          BNE     9$              ;JUST DATA
4199 025410                                  ERRDF   36.,EM47,ERR10  ;BAD DATA
     025410  104455                          TRAP    C$ERDF
     025412  000044                          .WORD   36
     025414  006604                          .WORD   EM47
     025416  010146                          .WORD   ERR10
4200 025420  005237  002234                  INC     CHECK           ;ACCOUNT FOR PRINT OF HEADER
4201 025424  000416                          BR      6$
4202
4203 025426                          9$:     PRINTB  #FRMT7,TMP1,GDDAT,BDDAT
     025426  013746  002302                  MOV     BDDAT,-(SP)
     025432  013746  002300                  MOV     GDDAT,-(SP)
     025436  013746  002274                  MOV     TMP1,-(SP)
     025442  012746  011354                  MOV     #FRMT7,-(SP)
     025446  012746  000004                  MOV     #4,-(SP)
     025452  010600                          MOV     SP,R0
     025454  104414                          TRAP    C$PNTB
```

```
     025456  062706  000012                    ADD     #12,SP
     025462                        6$:         CKLOOP
     025462  104406                            TRAP    C$CLP1

     025464  062737  000002  002276            ADD     #2,TMP2         ;NEXT
     025472  005237  002274                    INC     TMP1            ;NEXT
     025476  005237  002300                    INC     GDDAT           ;NEXT
     025502  023727  002274  000201            CMP     TMP1,#129.      ;DONE?
     025510  001315                            BNE     4$

     025512                                    ENDSEG                          ;**END OF SEGMENT**
     025512                        10001$:
     025512  104405                            TRAP    C$ESEG

     025514                                    ENDSEG                          ;**END OF SEGMENT**
     025514                        10000$:
     025514  104405                            TRAP    C$ESEG
     025516                        ENDTST                                      ;**END OF TEST**
     025516                        L10065:
     025516  104401                            TRAP    C$ETST


                                   .SBTTL  **TEST 28** - CHECK ZERO FILL ON WRITE

     025520                        BGNTST                                      ;**START OF TEST**

     025520                        STARS
                                   ;;*************************************************************
                                   ;WHEN WRITING PARTIAL SECTORS (LESS THAN 128 WORDS) THE
                                   ;CONTROLLER WILL FILL IN THE REMAINING PORTION OF
                                   ;THE SECTOR WITH ZERO WORDS.  CHECK THIS FEATURE
                                   ;WITH WORD COUNTS FROM 1 TO 127
     025520                        STARS
                                   ;;*************************************************************

     025520  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
     025524                                    CKERFG                  ;HEADS GO HOME OKAY
     025532  104432                            TRAP    C$EXIT
     025534  000442                            .WORD   L10066-.

     025536                                    BGNSEG                          ;**START OF SEGMENT**
     025536  104404                            TRAP    C$BSEG

     025540  012737  000001  002274            MOV     #1,TMP1         ;START WITH 1 WORD WRITE
     025546  012700  003426        35$:        MOV     #BUF,R0         ;WRITE BUFFER WITH 52525, WE'LL
     025552  012701  000200                    MOV     #128.,R1        ;WRITE 128 WORDS ALL THOUGH WE'RE
     025556  012720  052525        3$:         MOV     #52525,(R0)+    ;ONLY GOING TO TRANSFER < 128
     025562  005301                            DEC     R1              ;DONE WITH BUFFER?
     025564  001374                            BNE     3$              ;NO, GO BACK
     025566  013700  002274        33$:        MOV     TMP1,R0         ;GET TRANSFER WORD COUNT
     025572  005400                            NEG     R0              ;NEGATE FOR RLMP
     025574  010077  154562                    MOV     R0,@RLMP        ;STORE WORD COUNT AWAY
     025600  012777  003426  154550            MOV     #BUF,@RLBA      ;SET UP RLBA
     025606  005077  154546                    CLR     @RLDA
     025612  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
     025616  000012                            WRITE                   ;WRITE IT
     025620  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR WRITE TO FINISH
     025624                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
```

TEST 28 - CHECK ZERO FILL ON WRITE

```
                025624  104410                          TRAP    C$ESCAPE
                025626  000346                          .WORD   10000$-.
     4248
     4249       025630  004537  014612                  JSR     R5,CHERR            ;CHECK CNTLR FOR ERRORS
     4250       025634                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                025634  104410                          TRAP    C$ESCAPE
                025636  000336                          .WORD   10000$-.
     4251       025640                                  BGNSEG                  ;**START OF SEGMENT**
                025640  104404                          TRAP    C$BSEG
     4252       025642  012700  003426                  MOV     #BUF,R0         ;WE'RE GOING TO OVERLAY BUFFER BEFORE
     4253       025646  012701  000200                  MOV     #128.,R1        ;READING IT BACK.
     4254       025652  012720  125252          18$:    MOV     #125252,(R0)+   ;OVERLAY IT WITH COMPLIMENT
     4255       025656  005301                          DEC     R1              ;DONE?
     4256       025660  001374                          BNE     18$             ;NO, KEEP GOING
     4257       025662  012777  003426  154466          MOV     #BUF,@RLBA      ;SET UP TO READ
     4258       025670  012777  177600  154464          MOV     #-128.,@RLMP    ;128 WORDS TO CHECK ZERO FILL
     4259       025676  005077  154456                  CLR     @RLDA           ;SECTOR
     4260       025702  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
     4261       025706  000014                          READ
     4262       025710  004537  015700                  JSR     R5,WTCRDY       ;WAIT TIL WE FINISH THE READ
     4263       025714                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                025714  104410                          TRAP    C$ESCAPE
                025716  000234                          .WORD   10001$-.
     4264
     4265       025720  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
     4266       025724  005737  002236                  TST     T.CRC           ;WAS ERROR A DCK??
     4267       025730  001003                          BNE     8$              ;YES,SEE IF WE A DUMP
     4268       025732                          10$:    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                025732  104410                          TRAP    C$ESCAPE
                025734  000216                          .WORD   10001$-.
     4269       025736  000404                          BR      99$             ;SKIP AROUND
     4270       025740  005737  012440          8$:     TST     T.DMP           ;DO WE STILL WANT TO CHECK IT
     4271       025744  001772                          BEQ     10$             ;NO
     4272       025746                                  CKLOOP                  ;YES, CHECK FOR LOOP FIRST
                025746  104406                          TRAP    C$CLP1
     4273       025750  005037  002242          99$:    CLR     CDCNT           ;CLEAR NUMBER WE'RE TO PRINT
     4274       025754  005037  002234                  CLR     CHECK           ;ALLOW HEADER ON FIRST PRINT
     4275       025760  013702  002274                  MOV     TMP1,R2         ;WORDS WRITTEN IN R2
     4276       025764  012701  000200                  MOV     #128.,R1        ;CHECK 128 WORDS
     4277
     4278       025770  012703  003426                  MOV     #BUF,R3         ;SET UP BUFFER BEGINNING
     4279       025774  005037  002276                  CLR     TMP2            ;ZERO WORD COUNT
     4280       026000  012737  052525  002300          MOV     #52525,GDDAT    ;SET UP EXPECTED
     4281       026006  011337  002302          4$:     MOV     (R3),BDDAT      ;GET WORD
     4282       026012  023737  002302  002300          CMP     BDDAT,GDDAT     ;IS WORD CORRECT?
     4283       026020  001441                          BEQ     12$             ;YES, GO CHECK COUNTS AND REPEAT
     4284
     4285       026022  023737  002242  012442          CMP     CDCNT,T.LMT             ;CHECKED ENOUGH??
     4286       026030  001002                          BNE     333$            ;NO
     4287       026032                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                026032  104410                          TRAP    C$ESCAPE
                026034  000116                          .WORD   10001$-.
     4288       026036  005237  002242          333$:   INC     CDCNT           ;ACCOUNT FOR IT
     4289
     4290       026042  005737  002234                  TST     CHECK           ;HEADER OR JUST DATA
     4291       026046  001007                          BNE     9$              ;JUST DATA
     4292       026050                                  ERRDF   37.,EM27,ERR12
```

TEST 28 - CHECK ZERO FILL ON WRITE

```
        026050  104455                          TRAP    C$ERDF
        026052  000045                          .WORD   37
        026054  005734                          .WORD   EM27
        026056  010272                          .WORD   ERR12
 4293   026060  005237  002234                  INC     CHECK           ;ACCOUNT FOR PRINT OF HEADER
 4294   026064  000417                          BR      12$
 4295
 4296   026066                          9$:     PRINTB  #FRMT9,TMP1,R3,GDDAT,BDDAT
        026066  013746  002302                  MOV     BDDAT,-(SP)
        026072  013746  002300                  MOV     GDDAT,-(SP)
        026076  010346                          MOV     R3,-(SP)
        026100  013746  002274                  MOV     TMP1,-(SP)
        026104  012746  011547                  MOV     #FRMT9,-(SP)
        026110  012746  000005                  MOV     #5,-(SP)
        026114  010600                          MOV     SP,R0
        026116  104414                          TRAP    C$PNTB
        026120  062706  000014                  ADD     #14,SP
 4297   026124                          12$:    CKLOOP
        026124  104406                          TRAP    C$CLP1
 4298   026126  005723                  6$:     TST     (R3)+
 4299   026130  005237  002276                  INC     TMP2
 4300   026134  005301                          DEC     R1              ;DONE ALL WORDS?
 4301   026136  001405                          BEQ     7$              ;EXIT TEST
 4302   026140  005302                          DEC     R2              ;DONE CHECKING NON-ZERO WORDS
 4303   026142  003321                          BGT     4$              ;NO, BRANCH BACK
 4304   026144  005037  002300                  CLR     GDDAT           ;YES, SET EXP'D AS ZERO
 4305   026150  000716                          BR      4$              ;BRANCH BACK
 4306
 4307   026152                          7$:                             ;EXIT TEST
 4308   026152                                  ENDSEG                          ;**END OF SEGMENT**
        026152                          10001$:
        026152  104405                          TRAP    C$ESEG
 4309
 4310   026154  005237  002274                  INC     TMP1
 4311   026160  023727  002274  000200          CMP     TMP1,#128.
 4312   026166  001402                          BEQ     34$
 4313   026170  000137  025546                  JMP     35$
 4314   026174                          34$:
 4315
 4316   026174                                  ENDSEG                          ;**END OF SEGMENT**
        026174                          10000$:
        026174  104405                          TRAP    C$ESEG
 4317   026176                          ENDTST                          ;**END OF TEST**
        026176                          L10066:
        026176  104401                          TRAP    C$ETST
 4318
 4319                                   .SBTTL  **TEST 29** - CHECK SECTOR BITS OF HEADER COMPARE
 4320
 4321   026200                          BGNTST                          ;**START OF TEST**
 4322
 4323   026200                          STARS
                                        ;;**************************************************************************
 4324                                   ;TEST THAT ALL SECTOR BITS OF HEADER WORD CAN COMPARE
 4325                                   ;UNIQUELY.  WE TESTED THE HEADER COMPARE LOGIC EARLIER
 4326                                   ;BUT THAT WAS NOT AN EXTENSIVE TEST OF THE SECTOR BITS.
 4327                                   ;THE TEST PROCEDURE IS TO WRITE EACH SECTOR OF TRACK
 4328                                   ;0 WITH THE SECTOR ADDRESS, THEN GO BACK AND READ
```

TEST 29 - CHECK SECTOR BITS OF HEADER COMPARE

```
                                        ;EACH SECTOR.  IF ANY SECTOR HAS ANY DATA THEN THAT
                                        ;WHICH WAS EXPECTED THEN WE HAVE AN ERROR
                                        ;ERROR PRINT OUT WILL GIVE SECTOR, EXPECTED AND RECEIVED
026200                                  STARS
                                        ;;*****************************************************************
                                                JSR     PC,HDHOME       ;HEADS OVER TRACK 0
026200  004737  015764                        CKERFG                  ;HEADS GO HOME OKAY
026204
026212  104432                                TRAP    C$EXIT
026214  000414                                .WORD   L10067-.

026216                                          BGNSEG                          ;%%START OF SEGMENT%%
026216  104404                                  TRAP    C$BSEG

026220  005037  002272                1$:       CLR     TMP0            ;CLEAR

026224                                          BGNSEG                          ;%%START OF SEGMENT%%
026224  104404                                  TRAP    C$BSEG

026226  012702  003426                199$:     MOV     #BUF,R2         ;WRITE A PATTERN FOR THE WRITE
026232  012701  000200                          MOV     #128.,R1        ;ONE SECTOR'S WORTH
026236  013722  002272                2$:       MOV     TMP0,(R2)+      ;WRITE IT
026242  005301                                  DEC     R1              ;DONE,
026244  001374                                  BNE     2$              ;IF NOT, GO BACK

026246  012777  177600  154106                  MOV     #-128.,@RLMP    ;ONE SECTOR WORD COUNT
026254  012777  003426  154074                  MOV     #BUF,@RLBA      ;WRITE FROM BUF
026262  013777  002272  154070                  MOV     TMP0,@RLDA      ;SECTOR
026270  004537  015054                          JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
026274  000012                                  WRITE
026276  004537  015700                          JSR     R5,WTCRDY       ;WAIT FOR WRITE TO FINISH
026302                                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
026302  104410                                  TRAP    C$ESCAPE
026304  000320                                  .WORD   10001$-.
026306  005237  002272                          INC     TMP0            ;NEXT SECTOR
026312  023727  002272  000050                  CMP     TMP0,#40.       ;ALL DONE?
026320  001342                                  BNE     199$            ;NO GO BACK
026322  005037  002272                          CLR     TMP0            ;CLEAR

026326                                          BGNSEG                          ;%%START OF SEGMENT%%
026326  104404                                  TRAP    C$BSEG

026330  012702  003426                98$:      MOV     #BUF,R2         ;CLEAR THE BUFFER FIRST
026334  012701  000200                          MOV     #128.,R1        ;128 WORDS
026340  005022                        3$:       CLR     (R2)+
026342  005301                                  DEC     R1
026344  001375                                  BNE     3$

026346  013777  002272  154004                  MOV     TMP0,@RLDA      ;GET SECTOR
026354  012777  003426  153774                  MOV     #BUF,@RLBA      ;SETUP BUS ADDRESS

026362  012777  177600  153772                  MOV     #-128.,@RLMP    ;READ A SECTOR
026370  004537  015054                          JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
026374  000014                                  READ
026376  004537  015700                          JSR     R5,WTCRDY
026402                                          ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
026402  104410                                  TRAP    C$ESCAPE
```

TEST 29 - CHECK SECTOR BITS OF HEADER COMPARE

```
      026404  000216                          .WORD   10002$-.
4377
4378  026406  004537  014612                  JSR     R5,CHERR            ;CHECK CNTLR FOR ERRORS
4379  026412  005737  002236                  TST     T.CRC               ;WAS ERROR A DCK??
4380  026416  001003                          BNE     8$                  ;YES,SEE IF WE A DUMP
4381  026420                          10$:    ESCAPE  SEG                 ;CHECK FOR FL:LOE, ELSE EXIT SEG
      026420  104410                          TRAP    C$ESCAPE
      026422  000200                          .WORD   10002$-.
4382  026424  000404                          BR      99$                 ;SKIP AROUND
4383  026426  005737  012440          8$:     TST     T.DMP               ;DO WE STILL WANT TO CHECK IT
4384  026432  001772                          BEQ     10$                 ;NO
4385  026434                                  CKLOOP                      ;YES, CHECK FOR LOOP FIRST
      026434  104406                          TRAP    C$CLP1
4386
4387                                  ;CHECK NOW TO SEE IF WE READ THE RIGHT SECTOR
4388
4389  026436  005037  002242          99$:    CLR     CDCNT               ;CLEAR NUMBER WE'RE TO PRINT
4390  026442  005037  002234                  CLR     CHECK               ;ALLOW HEADER ON FIRST PRINT
4391  026446  013737  002272  002300          MOV     TMP0,GDDAT          ;EXPECTED DATA
4392  026454  012702  003426                  MOV     #BUF,R2             ;BUFFER
4393  026460  012701  000200                  MOV     #128.,R1            ;WORD COUNT
4394  026464  012237  002302          5$:     MOV     (R2)+,BDDAT         ;
4395  026470  023737  002302  002300          CMP     BDDAT,GDDAT
4396  026476  001440                          BEQ     6$
4397
4398  026500  023737  002242  012442          CMP     CDCNT,T.LMT                 ;CHECKED ENOUGH??
4399  026506  001002                          BNE     333$                ;NO
4400  026510                                  ESCAPE  SEG                 ;CHECK FOR FL:LOE, ELSE EXIT SEG
      026510  104410                          TRAP    C$ESCAPE
      026512  000110                          .WORD   10002$-.
4401  026514  005237  002242          333$:   INC     CDCNT               ;ACCOUNT FOR IT
4402
4403  026520  005737  002234                  TST     CHECK               ;HEADER OR JUST DATA
4404  026524  001007                          BNE     9$                  ;JUST DATA
4405  026526                                  ERRDF   38.,EM50,ERR11      ;
      026526  104455                          TRAP    C$ERDF
      026530  000046                          .WORD   38
      026532  006621                          .WORD   EM50
      026534  010220                          .WORD   ERR11
4406  026536  005237  002234                  INC     CHECK               ;ACCOUNT FOR PRINT OF HEADER
4407  026542  000416                          BR      6$
4408
4409  026544                          9$:     PRINTB  #FRMT8,TMP0,GDDAT,BDDAT
      026544  013746  002302                  MOV     BDDAT,-(SP)
      026550  013746  002300                  MOV     GDDAT,-(SP)
      026554  013746  002272                  MOV     TMP0,-(SP)
      026560  012746  011426                  MOV     #FRMT8,-(SP)
      026564  012746  000004                  MOV     #4,-(SP)
      026570  010600                          MOV     SP,R0
      026572  104414                          TRAP    C$PNTB
      026574  062706  000012                  ADD     #12,SP
4410  026600                          6$:     CKLOOP
      026600  104406                          TRAP    C$CLP1
4411
4412  026602  005301                          DEC     R1                  ;ALL OF SECTOR CHECKED?
4413  026604  001327                          BNE     5$                  ;GO BACK IF NOT
4414  026606  005237  002272                  INC     TMP0                ;NEXT SECTOR
```

```
**TEST 29** - CHECK SECTOR BITS OF HEADER COMPARE

4415 026612  023727  002272  000050          CMP     TMP0,#40.       ;DONE?
4416 026620  001243                          BNE     98$             ;NO, GO BACK
4417
4418 026622                                  ENDSEG                          ;%%END OF SEGMENT%%
     026622                          10002$:
     026622  104405                        TRAP    C$ESEG
4419
4420 026624                                  ENDSEG                          ;%%END OF SEGMENT%%
     026624                          10001$:
     026624  104405                        TRAP    C$ESEG
4421 026626                                  ENDSEG                          ;%%END OF SEGMENT%%
     026626                          10000$:
     026626  104405                        TRAP    C$ESEG
4422 026630                          ENDTST                                  ;**END OF TEST**
     026630                          L10067:
     026630  104401                        TRAP    C$ETST
4423
4424                                 .SBTTL  **TEST 30** - WRITE CHECK NPR INTEGRITY
4425
4426 026632                          BGNTST                                  ;**START OF TEST**
4427
4428 026632                          STARS
                                     ;;**********************************************************************
4429                                 ;CHECK THAT NPR WILL NOT INTERFERE WITH THE OPERATION OF THE
4430                                 ;UNIBUS. WE SET UP LOCATION 4 TO HANDLE THE TRAP IF IT HAPPENS.
4431 026632                          STARS
                                     ;;**********************************************************************
4432
4433 026632  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
4434 026636                                  CKERFG                  ;HEADS GO HOME OKAY
     026644  104432                          TRAP    C$EXIT
     026646  000376                          .WORD   L10070-.
4435
4436 026650                                  BGNSEG                          ;%%START OF SEGMENT%%
     026650  104404                          TRAP    C$BSEG
4437
4438 026652  012700  003426                  MOV     #BUF,R0         ;SETUP AND WRITE
4439 026656  012701  000200                  MOV     #128.,R1        ;128 WORDS
4440 026662  012720  125252          299$:   MOV     #125252,(R0)+   ;WRITE
4441 026666  005301                          DEC     R1              ;DONE??
4442 026670  001374                          BNE     299$
4443
4444 026672  012777  003426  153456          MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
4445 026700  012777  177600  153454          MOV     #-128.,@RLMP    ;WORD COUNT
4446 026706  005077  153446                  CLR     @RLDA           ;CLEAR DISK ADDRESS
4447 026712  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4448 026716  000012                          WRITE
4449 026720  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
4450 026724                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     026724  104410                          TRAP    C$ESCAPE
     026726  000314                          .WORD   10000$-.
4451 026730  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4452 026734                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     026734  104410                          TRAP    C$ESCAPE
     026736  000304                          .WORD   10000$-.
4453
4454                                 ;VERIFY WRITE WITH READ BEFORE WRCHK
```

TEST 30 - WRITE CHECK NPR INTEGRITY

```
4455
4456 026740  005077  153414                    CLR     @RLDA
4457 026744  012777  003426  153404            MOV     #BUF,@RLBA
4458 026752  012777  177600  153402            MOV     #-128.,@RLMP
4459 026760  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4460 026764  000014                            READ
4461 026766  004537  015700                    JSR     R5,WTCRDY
4462 026772                                    ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     026772  104410                            TRAP    C$ESCAPE
     026774  000246                            .WORD   10000$-.
4463 026776  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4464 027002                                    ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027002  104410                            TRAP    C$ESCAPE
     027004  000236                            .WORD   10000$-.
4465
4466 027006                                    BGNSEG                          ;%%START OF SEGMENT%%
     027006  104404                            TRAP    C$BSEG
4467
4468 027010                            1$:     SETVEC  ERRVEC,#TRPHAN,#340     ;SET UP FOR TRAP
     027010  012746  000340                    MOV     #340,-(SP)
     027014  012746  015756                    MOV     #TRPHAN,-(SP)
     027020  013746  002244                    MOV     ERRVEC,-(SP)
     027024  012746  000003                    MOV     #3,-(SP)
     027030  104437                            TRAP    C$SVEC
     027032  062706  000010                    ADD     #10,SP
4469 027036  005037  002254                    CLR     TRPFLG                  ;CLEAR TRAP OCCURANCE
4470 027042  012777  003426  153306            MOV     #BUF,@RLBA              ;BUS ADDRESS
4471 027050  005077  153304                    CLR     @RLDA                   ;LOAD DISK ADDRESS
4472 027054  012777  177600  153300            MOV     #-128.,@RLMP            ;WORD COUNT OF 128
4473 027062  005037  002300                    CLR     GDDAT                   ;SET UP CSR TO LOAD
4474 027066  013737  002246  002300            MOV     DRIVE,GDDAT             ;SET IN DRIVE
4475 027074  052737  000002  002300            BIS     #WRCHK,GDDAT            ;SET IN FUNCTION
4476 027102  004537  015362                    JSR     R5,BEFORE               ;LOAD FOR ERROR PRINTOUT
4477 027106  013737  002300  002330            MOV     GDDAT,B.CS              ;SET IN COMMAND
4478 027114  052737  000201  002330            BIS     #201,B.CS               ;LOAD CRDY
4479 027122  042737  002000  002330            BIC     #OPI,B.CS               ;CLEAR (BIT 10)
4480 027130  013777  002300  153216            MOV     GDDAT,@RLCS             ;ISSUE WRITE CHECK
4481 027136  012701  000144                    MOV     #100.,R1                ;WAIT FOR CRDY
4482 027142  032777  000200  153204    5$:     BIT     #CRDY,@RLCS             ;NPR DONE
4483 027150  001015                            BNE     6$                      ;YES, 6$
4484 027152                                    WAITUS  #20.                    ;WAIT A WHILE
4485 027164  005301                            DEC     R1                      ;A WHILE UP
4486 027166  001365                            BNE     5$                      ;NO, GO BACK
4487
4488 027170  004537  015414                    JSR     R5,AFTER
4489 027174                                    ERRDF   0.,CRTIM,ERR5           ;CONTROLLER TIMED OUT
     027174  104455                            TRAP    C$ERDF
     027176  000000                            .WORD   0
     027200  003521                            .WORD   CRTIM
     027202  007722                            .WORD   ERR5
4490 027204                            6$:     CLRVEC  ERRVEC                  ;CLEAR VECTOR
     027204  013700  002244                    MOV     ERRVEC,R0
     027210  104436                            TRAP    C$CVEC
4491 027212                                    ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027212  104410                            TRAP    C$ESCAPE
     027214  000024                            .WORD   10001$-.
4492
```

```
**TEST 30** - WRITE CHECK NPR INTEGRITY

4493 027216  005737  002254                    TST     TRPFLG                  ;DID TRAP OCCUR?
4494 027222  001406                            BEQ     7$                      ;NO
4495 027224  004537  015414                    JSR     R5,AFTER
4496 027230                                    ERRSF   1.,EM57,ERRO            ;TRAP ON WRITE
     027230  104454                            TRAP    C$ERSF
     027232  000001                            .WORD   1
     027234  007052                            .WORD   EM57
     027236  007510                            .WORD   ERRO
4497 027240                             7$:
4498
4499 027240                                    ENDSEG                          ;%%END OF SEGMENT%%
     027240                             10001$:
     027240  104405                            TRAP    C$ESEG
4500 027242                                    ENDSEG                          ;%%END OF SEGMENT%%
     027242                             10000$:
     027242  104405                            TRAP    C$ESEG
4501
4502 027244                             ENDTST                                 ;**END OF TEST**
     027244                             L10070:
     027244  104401                            TRAP    C$ETST
4503
4504                                    .SBTTL  **TEST 31** - WRITE CHECK FUNCTION
4505
4506 027246                             BGNTST                                 ;**START OF TEST**
4507
4508 027246                             STARS
                                        ;;**********************************************************************
4509                                    ;CHECK OF WRITE CHECK LOGIC UNDER FLAG MODE
4510                                    ; WE WILL WRITE CHECK A FULL SECTOR (128 WORDS) FROM
4511                                    ;MEMORY (BUF).  WE CHECK THAT NO ERRORS OCCUR.
4512 027246                             STARS
                                        ;;**********************************************************************
4513
4514 027246  004737  015764                    JSR     PC,HDHOME               ;HEADS OVER TRACK 0
4515 027252                                    CKERFG                          ;HEADS GO HOME OKAY
     027260  104432                            TRAP    C$EXIT
     027262  000214                            .WORD   L10071-.
4516
4517 027264                                    BGNSEG                          ;%%START OF SEGMENT%%
     027264  104404                            TRAP    C$BSEG
4518
4519 027266  012700  003426                    MOV     #BUF,R0                 ;SETUP AND WRITE
4520 027272  012701  000200                    MOV     #128.,R1                ;128 WORDS
4521 027276  012720  125252             299$:  MOV     #125252,(R0)+           ;WRITE
4522 027302  005301                            DEC     R1                      ;DONE??
4523 027304  001374                            BNE     299$
4524
4525 027306  012777  003426  153042            MOV     #BUF,@RLBA              ;LOAD BUS ADDRESS
4526 027314  012777  177600  153040            MOV     #-128.,@RLMP            ;WORD COUNT
4527 027322  005077  153032                    CLR     @RLDA                   ;CLEAR DISK ADDRESS
4528 027326  004537  015054                    JSR     R5,LDFUNC                       ;LOAD THE FUNCTION IN NEXT WORD
4529 027332  000012                            WRITE
4530 027334  004537  015700                    JSR     R5,WTCRDY               ;WAIT FOR CONTROLLER READY
4531 027340                                    ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027340  104410                            TRAP    C$ESCAPE
     027342  000132                            .WORD   10000$-.
4532 027344  004537  014612                    JSR     R5,CHERR                        ;CHECK CNTLR FOR ERRORS
```

TEST 31 - WRITE CHECK FUNCTION

```
4533 027350                            ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027350  104410                    TRAP    C$ESCAPE
     027352  000122                    .WORD   10000$-.
4534 027354                            BGNSEG                  ;%%START OF SEGMENT%%
     027354  104404                    TRAP    C$BSEG
4535
4536                                ;VERIFY WRITE WITH READ BEFORE WRCHK
4537
4538 027356  005077  152776            CLR     @RLDA
4539 027362  012777  003426  152766    MOV     #BUF,@RLBA
4540 027370  012777  177600  152764    MOV     #-128.,@RLMP
4541 027376  004537  015054            JSR     R5,LDFUNC       ;LOAD THE FUNCTION IN NEXT WORD
4542 027402  000014                    READ
4543 027404  004537  015700            JSR     R5,WTCRDY
4544 027410                            ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027410  104410                    TRAP    C$ESCAPE
     027412  000060                    .WORD   10001$-.
4545 027414  004537  014612            JSR     R5,CHERR        ;CHECK CNTLR FOR ERRORS
4546 027420                            ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027420  104410                    TRAP    C$ESCAPE
     027422  000050                    .WORD   10001$-.
4547
4548 027424                            BGNSEG                  ;%%START OF SEGMENT%%
     027424  104404                    TRAP    C$BSEG
4549
4550 027426                         3$:
4551 027426  005077  152726            CLR     @RLDA
4552 027432  012777  177600  152722    MOV     #-128.,@RLMP    ;WORD COUNT
4553 027440  012777  003426  152710    MOV     #BUF,@RLBA      ;BUS ADDRESS
4554 027446  004537  015054            JSR     R5,LDFUNC       ;LOAD THE FUNCTION IN NEXT WORD
4555 027452  000002                    WRCHK                   ;WRITE CHECK
4556
4557 027454  004537  015700            JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
4558 027460                            ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027460  104410                    TRAP    C$ESCAPE
     027462  000006                    .WORD   10002$-.
4559
4560 027464  004537  014612            JSR     R5,CHERR        ;CHECK CNTLR FOR ERRORS
4561
4562 027470                            ENDSEG                  ;%%END OF SEGMENT%%
     027470                         10002$:
     027470  104405                    TRAP    C$ESEG
4563 027472                            ENDSEG                  ;%%END OF SEGMENT%%
     027472                         10001$:
     027472  104405                    TRAP    C$ESEG
4564 027474                            ENDSEG                  ;%%END OF SEGMENT%%
     027474                         10000$:
     027474  104405                    TRAP    C$ESEG
4565 027476                         ENDTST                     ;**END OF TEST**
     027476                         L10071:
     027476  104401                    TRAP    C$ETST
4566
4567                                .SBTTL  **TEST 32** - WRITE CHECK FUNCTION INTERRUPT
4568
4569 027500                         BGNTST                     ;**START OF TEST**
4570
4571 027500                         STARS
```

TEST 32 - WRITE CHECK FUNCTION INTERRUPT

```
                                        ;;*****************************************************************
                                        ;CHECK OF WRITE CHECK LOGIC UNDER INTERRUPT MODE
                                        ;WE WILL WRITE CHECK A FULL SECTOR (128 WORDS) FROM MEMORY (BUF).
                                        ;WE CHECK THAT NO ERRORS OCCUR.  WE DO NOT CHECK RLDA OR RLBA
                                        ;INCREMENT AT THIS TIME.
4576 027500                             STARS
                                        ;;*****************************************************************
4577
4578 027500  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
4579 027504                                    CKERFG                  ;HEADS GO HOME OKAY
     027512  104432                            TRAP    C$EXIT
     027514  000252                            .WORD   L10072-.
4580
4581 027516                                    BGNSEG                          ;**START OF SEGMENT**
     027516  104404                            TRAP    C$BSEG
4582
4583 027520  012700  003426                    MOV     #BUF,R0         ;SETUP AND WRITE
4584 027524  012701  000200                    MOV     #128.,R1        ;128 WORDS
4585 027530  012720  125252          299$:     MOV     #125252,(R0)+   ;WRITE
4586 027534  005301                            DEC     R1              ;DONE??
4587 027536  001374                            BNE     299$
4588
4589 027540  012777  003426  152610            MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
4590 027546  012777  177600  152606            MOV     #-128.,@RLMP    ;WORD COUNT
4591 027554  005077  152600                    CLR     @RLDA           ;CLEAR DISK ADDRESS
4592 027560  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4593 027564  000012                            WRITE
4594 027566  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
4595 027572                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027572  104410                            TRAP    C$ESCAPE
     027574  000170                            .WORD   10000$-.
4596 027576  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4597 027602                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027602  104410                            TRAP    C$ESCAPE
     027604  000160                            .WORD   10000$-.
4598                                    ;VERIFY WRITE WITH READ BEFORE WRCHK
4599
4600 027606  005077  152546                    CLR     @RLDA
4601 027612  012777  003426  152536            MOV     #BUF,@RLBA
4602 027620  012777  177600  152534            MOV     #-128.,@RLMP
4603 027626  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4604 027632  000014                            READ
4605 027634  004537  015700                    JSR     R5,WTCRDY
4606 027640                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027640  104410                            TRAP    C$ESCAPE
     027642  000122                            .WORD   10000$-.
4607 027644  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4608 027650                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     027650  104410                            TRAP    C$ESCAPE
     027652  000112                            .WORD   10000$-.
4609
4610 027654                                    BGNSEG                          ;**START OF SEGMENT**
     027654  104404                            TRAP    C$BSEG
4611
4612 027656  005037  002256                    CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE FLAG
4613 027662  005077  152472                    CLR     @RLDA
4614 027666  012777  177600  152466            MOV     #-128.,@RLMP    ;SET UP WORD COUNT
```

```
**TEST 32** - WRITE CHECK FUNCTION INTERRUPT

 4615 027674  012777  003426  152454          MOV     #BUF,@RLBA      ;SET UP BUS ADDRESS
 4616
 4617 027702                                  SETPRI  #PRI00          ;PRIORITY TO 0
      027702  012700  000000                  MOV     #PRI00,R0
      027706  104441                          TRAP    C$SPRI
 4618 027710  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 4619 027714  000102                          WRCHK!INTEN             ;WRITE CHECK UNDER INTERRUPT
 4620 027716  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR INTERRUPT
 4621 027722                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      027722  104410                          TRAP    C$ESCAPE
      027724  000036                          .WORD   10001$-.
 4622
 4623                                 ;       SETPRI  #PRI07          ;SET PRIORITY TO 7              ;JSD REV A
 4624 027726                                  SETPRI  #PRI06          ;SET PRIORITY TO 6              ;JSD REV A
      027726  012700  000300                  MOV     #PRI06,R0
      027732  104441                          TRAP    C$SPRI
 4625 027734  005737  002256                  TST     INTFLG          ;DID INTERRUPT OCCUR?
 4626 027740  001004                          BNE     2$              ;YES-BRANCH NO-REPORT
 4627
 4628 027742                                  ERRDF   4.,EM60,ERR0    ;WRITE DID NOT INTERRUPT
      027742  104455                          TRAP    C$ERDF
      027744  000004                          .WORD   4
      027746  007107                          .WORD   EM60
      027750  007510                          .WORD   ERR0
 4629 027752                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      027752  104410                          TRAP    C$ESCAPE
      027754  000006                          .WORD   10001$-.
 4630
 4631 027756  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 4632
 4633 027762                                  ENDSEG                          ;**END OF SEGMENT**
      027762                          10001$:
      027762  104405                          TRAP    C$ESEG
 4634 027764                                  ENDSEG                          ;**END OF SEGMENT**
      027764                          10000$:
      027764  104405                          TRAP    C$ESEG
 4635 027766                          ENDTST                                  ;**END OF TEST**
      027766                          L10072:
      027766  104401                          TRAP    C$ETST
 4636
 4637                                 .SBTTL  **TEST 33** - PROPER INCREMENT OF RLBA ON WRITE CHECK
 4638
 4639 027770                          BGNTST                                  ;**START OF TEST**
 4640
 4641 027770                          STARS
                                      ;;**********************************************************************
 4642                                 ;CHECK THAT THE RLBA WILL INCREMENT PROPERLY AFTER THE
 4643                                 ;WRITE CHECK WAS FINISHED THE RLBA SHOULD BE 128 WORDS (256 BYTES)
 4644                                 ;CREATER.  STARTING RLBA IS "BUF", ENDING SHOULD BE "BUF + 256."
 4645                                 ;WE WILL MONITOR ALL ERRORS AND REPORT THEM ACCORDINGLY
 4646 027770                          STARS
                                      ;;**********************************************************************
 4647
 4648 027770  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 4649 027774                                  CKERFG                  ;HEADS GO HOME OKAY
      030002  104432                          TRAP    C$EXIT
      030004  000256                          .WORD   L10073-.
```

TEST 33 - PROPER INCREMENT OF RLBA ON WRITE CHECK

```
4650
4651 030006                                  BGNSEG                          ;**START OF SEGMENT**
     030006  104404                          TRAP    C$BSEG
4652
4653 030010  012700  003426                  MOV     #BUF,R0         ;SETUP AND WRITE
4654 030014  012701  000200                  MOV     #128.,R1        ;128 WORDS
4655 030020  012720  125252          299$:   MOV     #125252,(R0)+   ;WRITE
4656 030024  005301                          DEC     R1              ;DONE??
4657 030026  001374                          BNE     299$
4658
4659 030030  012777  003426  152320          MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
4660 030036  012777  177600  152316          MOV     #-128.,@RLMP    ;WORD COUNT
4661 030044  005077  152310                  CLR     @RLDA           ;CLEAR DISK ADDRESS
4662 030050  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4663 030054  000012                          WRITE
4664 030056  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
4665 030062                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030062  104410                          TRAP    C$ESCAPE
     030064  000174                          .WORD   10000$-.
4666 030066  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4667 030072                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030072  104410                          TRAP    C$ESCAPE
     030074  000164                          .WORD   10000$-.
4668                                 ;VERIFY WRITE WITH READ BEFORE WRCHK
4669
4670 030076  005077  152256                  CLR     @RLDA
4671 030102  012777  003426  152246          MOV     #BUF,@RLBA
4672 030110  012777  177600  152244          MOV     #-128.,@RLMP
4673 030116  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4674 030122  000014                          READ
4675 030124  004537  015700                  JSR     R5,WTCRDY
4676 030130                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030130  104410                          TRAP    C$ESCAPE
     030132  000126                          .WORD   10000$-.
4677 030134  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4678 030140                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030140  104410                          TRAP    C$ESCAPE
     030142  000116                          .WORD   10000$-.
4679
4680 030144                                  BGNSEG                          ;**START OF SEGMENT**
     030144  104404                          TRAP    C$BSEG
4681
4682 030146                          3$:
4683 030146  005077  152206                  CLR     @RLDA
4684 030152  012777  003426  152176          MOV     #BUF,@RLBA      ;SET UP BUS ADDRESS
4685 030160  012777  177600  152174          MOV     #-128.,@RLMP    ;WORD COUNT
4686 030166  012737  003426  002300          MOV     #BUF,@DDAT      ;FORM EXPECTED BUS ADDRESS
4687 030174  062737  000400  002300          ADD     #256.,@DDAT     ;AFTER WRITE
4688
4689 030202  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4690 030206  000002                          WRCHK                   ;WRITE CHECK
4691 030210  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
4692 030214                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030214  104410                          TRAP    C$ESCAPE
     030216  000040                          .WORD   10001$-.
4693
4694 030220  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
```

```
**TEST 33** - PROPER INCREMENT OF RLBA ON WRITE CHECK

4695 030224                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030224  104410                        TRAP    C$ESCAPE
     030226  000030                        .WORD   10001$-.
4696 030230  017737  152122  002302        MOV     @RLBA,BDDAT     ;READ 'RLBA' FOR PRESENT ADDRESS
4697 030236  023737  002302  002300        CMP     BDDAT,GDDAT     ;DID 'BA' INCREMENT PROPERLY?
4698 030244  001404                        BEQ     2$              ;YES, CONTINUE
4699
4700 030246                                ERRDF   5.,EM61,ERR4    ;BA DID NOT INCREMENT
     030246  104455                        TRAP    C$ERDF
     030250  000005                        .WORD   5
     030252  007137                        .WORD   EM61
     030254  007654                        .WORD   ERR4
4701
4702 030256                           2$:
4703
4704 030256                                ENDSEG                          ;**END OF SEGMENT**
     030256                           10001$:
     030256  104405                        TRAP    C$ESEG
4705 030260                                ENDSEG                          ;**END OF SEGMENT**
     030260                           10000$:
     030260  104405                        TRAP    C$ESEG
4706 030262                           ENDTST                               ;**END OF TEST**
     030262                           L10073:
     030262  104401                        TRAP    C$ETST
4707
4708                                  .SBTTL  **TEST 34** - PROPER INCREMENT OF RLDA ON WRITE CHECK
4709
4710 030264                           BGNTST                               ;**START OF TEST**
4711
4712 030264                           STARS
                                      ;;*************************************************************************
4713                                  ;CHECK THAT THE SECTOR INCREMENTS AFTER THE WRITE CHECK WAS FINISHED.
4714                                  ;A FULL SECTOR WRITE CHECK THE RLDA SHOULD REFLECT AN INCREMENT
4715                                  ;OF THE SECOTR.  "GDDAT" WAS THE EXPECTED RLDA.
4716 030264                           STARS
                                      ;;*************************************************************************
4717
4718 030264  004737  015764                JSR     PC,MDHOME       ;HEADS OVER TRACK 0
4719 030270                                CKERFG                  ;HEADS GO HOME OKAY
     030276  104432                        TRAP    C$EXIT
     030300  000254                        .WORD   L10074-.
4720
4721 030302                                BGNSEG                          ;**START OF SEGMENT**
     030302  104404                        TRAP    C$BSEG
4722
4723 030304  012700  003426                MOV     #BUF,R0         ;SETUP AND WRITE
4724 030310  012701  000200                MOV     #128.,R1        ;128 WORDS
4725 030314  012720  125252          299$: MOV     #125252,(R0)+   ;WRITE
4726 030320  005301                        DEC     R1              ;DONE??
4727 030322  001374                        BNE     299$
4728
4729 030324  012777  003426  152024        MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
4730 030332  012777  177600  152022        MOV     #-128.,@RLMP    ;WORD COUNT
4731 030340  005077  152014                CLR     @RLDA           ;CLEAR DISK ADDRESS
4732 030344  004537  015054                JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4733 030350  000012                        WRITE
4734 030352  004537  015700                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
```

TEST 34 - PROPER INCREMENT OF RLDA ON WRITE CHECK

```
4735 030356                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030356  104410                        TRAP    C$ESCAPE
     030360  000172                        .WORD   10000$-.
4736 030362  004537  014612                JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4737 030366                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030366  104410                        TRAP    C$ESCAPE
     030370  000162                        .WORD   10000$-.
4738                               ;VERIFY WRITE WITH READ BEFORE WRCHK
4739
4740 030372  005077  151762                CLR     @RLDA
4741 030376  012777  003426  151752        MOV     #BUF,@RLBA
4742 030404  012777  177600  151750        MOV     #-128.,@RLMP
4743 030412  004537  015054                JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4744 030416  000014                        READ
4745 030420  004537  015700                JSR     R5,WTCRDY
4746 030424                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030424  104410                        TRAP    C$ESCAPE
     030426  000124                        .WORD   10000$-.
4747 030430  004537  014612                JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4748 030434                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030434  104410                        TRAP    C$ESCAPE
     030436  000114                        .WORD   10000$-.
4749
4750 030440                                BGNSEG                          ;%%START OF SEGMENT%%
     030440  104404                        TRAP    C$BSEG
4751
4752 030442                        3$:
4753 030442  005037  002300                CLR     GDDAT
4754 030446  013777  002300  151704        MOV     GDDAT,@RLDA     ;SETUP DISK ADDRESS
4755 030454  005237  002300                INC     GDDAT           ;CREATE EXPECTED SECTOR
4756 030460  012777  177600  151674        MOV     #-128.,@RLMP    ;WORD COUNT
4757 030466  012777  003426  151662        MOV     #BUF,@RLBA      ;SETUP BUS ADDRESS
4758
4759 030474  004537  015054                JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4760 030500  000002                        WRCHK                   ;WRITE CHECK
4761 030502  004537  015700                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
4762 030506                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030506  104410                        TRAP    C$ESCAPE
     030510  000040                        .WORD   10001$-.
4763
4764 030512  004537  014612                JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4765 030516                                ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030516  104410                        TRAP    C$ESCAPE
     030520  000030                        .WORD   10001$-.
4766
4767 030522  013737  002344  002302        MOV     E.DA,BDDAT      ;READ DISK ADDRESS
4768 030530  023737  002300  002302        CMP     GDDAT,BDDAT     ;DID SECTOR INCREMENT PROPERLY
4769 030536  001404                        BEQ     2$              ;YES, BRANCH  NO, REPORT ERROR
4770
4771 030540                                ERRDF   6.,EM62,ERR4    ;DA DID NOT INCREMENT
     030540  104455                        TRAP    C$ERDF
     030542  000006                        .WORD   6
     030544  007207                        .WORD   EM62
     030546  007654                        .WORD   ERR4
4772
4773 030550                        2$:
4774
```

```
**TEST 34** - PROPER INCREMENT OF RLDA ON WRITE CHECK

                                        ENDSEG                          ;##END OF SEGMENT##
                                10001$:
                                        TRAP    C$ESEG
                                        ENDSEG                          ;##END OF SEGMENT##
                                10000$:
                                        TRAP    C$ESEG
                                ENDTST                                  ;**END OF TEST**
                                L10074:
                                        TRAP    C$ETST

                                .SBTTL  **TEST 35** - MULTIPLE SECTOR WRITE CHECK

                                BGNTST                                  ;**START OF TEST**

                                STARS
                                ;;**************************************************************************
                                ;CHECK FOR MULTIPLE SECTOR WRITE CHECK. THIS TEST CHECKS
                                ;THAT TWO SECTORS CAN BE SUCCESSFULLY CHECKED.  WE LOAD
                                ;A WORD COUNT OF 129 WORDS (ONE SECTOR + 1 WORD) STARTING AT
                                ;SECTOR 0 THRU SECTOR 37 AND VERIFY THAT THE RLDA DOES
                                ;A DOUBLE INCREMENT EACH TIME.
                                STARS
                                ;;**************************************************************************

                                        JSR     PC,HDHOME       ;HEADS OVER TRACK 0
                                        CKERFG                  ;HEADS GO HOME OKAY
                                        TRAP    C$EXIT
                                        .WORD   L10075-.

                                        BGNSEG                          ;##START OF SEGMENT##
                                        TRAP    C$BSEG

                                        MOV     #0,TMP0
                                        MOV     #0,TMP1
                                        MOV     #BUF,R0         ;SETUP AND WRITE
                                        MOV     #129.,R1        ;129 WORDS
                                299$:   MOV     #125252,(R0)+   ;WRITE
                                        DEC     R1              ;DONE??
                                        BNE     299$

                                1$:     MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
                                        MOV     #-129.,@RLMP    ;WORD COUNT
                                        MOV     TMP1,GDDAT
                                        BIS     TMP0,GDDAT
                                        MOV     GDDAT,@RLDA
                                        JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
                                        WRITE
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
                                        ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                                        TRAP    C$ESCAPE
                                        .WORD   10000$-.
                                        JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
                                        ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                                        TRAP    C$ESCAPE
                                        .WORD   10000$-.

                                ;VERIFY WRITE WITH READ BEFORE WRCHK
```

TEST 35 - MULTIPLE SECTOR WRITE CHECK

```
4817
4818 030716  013737  002274  002300          MOV     TMP1,GDDAT
4819 030724  053737  002272  002300          BIS     TMP0,GDDAT
4820 030732  013777  002300  151420          MOV     GDDAT,@RLDA
4821 030740  012777  003426  151410          MOV     #BUF,@RLBA
4822 030746  012777  177577  151406          MOV     #-129.,@RLMP
4823 030754  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4824 030760  000014                          READ
4825 030762  004537  015700                  JSR     R5,WTCRDY
4826 030766                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030766  104410                          TRAP    C$ESCAPE
     030770  000154                          .WORD   10000$-.
4827 030772  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4828 030776                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     030776  104410                          TRAP    C$ESCAPE
     031000  000144                          .WORD   10000$-.
4829
4830 031002                                  BGNSEG                          ;**START OF SEGMENT**
     031002  104404                          TRAP    C$BSEG
4831
4832 031004  013737  002274  002300          MOV     TMP1,GDDAT      ;GET CYLINDER
4833 031012  053737  002272  002300          BIS     TMP0,GDDAT      ;GET SECTOR
4834 031020  013777  002300  151332          MOV     GDDAT,@RLDA     ;SET DISK ADDRESS-SECTOR 0
4835 031026  062737  000002  002300          ADD     #2,GDDAT        ;SET EXPECTED + 2
4836 031034  012777  003426  151314          MOV     #BUF,@RLBA      ;SET BUS ADDRESS
4837 031042  012777  177577  151312          MOV     #-129.,@RLMP    ;WORD COUNT-SECTOR+1 WORD
4838
4839 031050  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4840 031054  000002                          WRCHK                   ;WRITE CHECK
4841 031056  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY?
4842 031062                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     031062  104410                          TRAP    C$ESCAPE
     031064  000042                          .WORD   10001$-.
4843
4844 031066  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4845 031072                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     031072  104410                          TRAP    C$ESCAPE
     031074  000032                          .WORD   10001$-.
4846
4847 031076  013737  002344  002302          MOV     E.DA,BDDAT      ;READ DISK ADDRESS
4848 031104  023737  002302  002300          CMP     BDDAT,GDDAT     ;IS DISK ADDRESS CORRECT
4849 031112  001404                          BEQ     2$              ;YES, BRANCH  NO, REPORT ERROR
4850
4851 031114                                  ERRDF   7.,EM63,ERR4    ;DISK ADDRESS NOT CORRECT
     031114  104455                          TRAP    C$ERDF
     031116  000007                          .WORD   7
     031120  007246                          .WORD   EM63
     031122  007654                          .WORD   ERR4
4852
4853 031124                          2$:     CKLOOP
     031124  104406                          TRAP    C$CLP1
4854
4855 031126                                  ENDSEG                          ;**END OF SEGMENT**
     031126                          10001$:
     031126  104405                          TRAP    C$ESEG
4856
4857 031130  005237  002272                  INC     TMP0            ;NEXT SECTOR
```

```
**TEST 35** - MULTIPLE SECTOR WRITE CHECK

 4858 031134  022737  000046  002272          CMP     #46,TMPO        ;AT END?
 4859 031142  001233                          BNE     1$              ;NO, GO BACK
 4860 031144                                  ENDSEG                          ;%%END OF SEGMENT%%
      031144                          10000$:
      031144  104405                        TRAP    C$ESEG
 4861 031146                          ENDTST                                  ;**END OF TEST**
      031146                          L10075:
      031146  104401                        TRAP    C$ETST
 4862                                 .SBTTL  **TEST 36** - FORCE DCK WITH WRITE CHECK
 4863
 4864 031150                          BGNTST                                  ;**START OF TEST**
 4865
 4866 031150                          STARS
                                      ;;**********************************************************************
 4867                                 ;FORCE A DCK WITH WRITE CHECK. THIS IS DONE BY WRITING
 4868                                 ;A SECTOR AND CHANGING A WORD IN MEMORY BEFORE WRITE CHECK
 4869                                 ;IS ISSUED..
 4870 031150                          STARS
                                      ;;**********************************************************************
 4871
 4872 031150  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 4873 031154                                  CKERFG                  ;HEADS GO HOME OKAY
      031162  104432                          TRAP    C$EXIT
      031164  000262                          .WORD   L10076-.
 4874
 4875 031166                                  BGNSEG                          ;%%START OF SEGMENT%%
      031166  104404                          TRAP    C$BSEG
 4876
 4877 031170  012700  003426                  MOV     #BUF,R0         ;SETUP AND WRITE
 4878 031174  012701  000200                  MOV     #128.,R1        ;128 WORDS
 4879 031200  012720  125252          299$:   MOV     #125252,(R0)+   ;WRITE
 4880 031204  005301                          DEC     R1              ;DONE??
 4881 031206  001374                          BNE     299$
 4882
 4883 031210  012777  003426  151140          MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
 4884 031216  012777  177600  151136          MOV     #-128.,@RLMP    ;WORD COUNT
 4885 031224  005077  151130                  CLR     @RLDA           ;CLEAR DISK ADDRESS
 4886 031230  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 4887 031234  000012                          WRITE
 4888 031236  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
 4889 031242                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      031242  104410                          TRAP    C$ESCAPE
      031244  000200                          .WORD   10000$-.
 4890 031246  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 4891 031252                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      031252  104410                          TRAP    C$ESCAPE
      031254  000170                          .WORD   10000$-.
 4892                                 ;VERIFY WRITE WITH READ BEFORE WRCHK
 4893
 4894 031256  005077  151076                  CLR     @RLDA
 4895 031262  012777  003426  151066          MOV     #BUF,@RLBA
 4896 031270  012777  177600  151064          MOV     #-128.,@RLMP
 4897 031276  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 4898 031302  000014                          READ
 4899 031304  004537  015700                  JSR     R5,WTCRDY
 4900 031310                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      031310  104410                          TRAP    C$ESCAPE
```

TEST 36 - FORCE DCK WITH WRITE CHECK

```
                031312  000132                          .WORD   10000$-.
4901            031314  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4902            031320                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                031320  104410                          TRAP    C$ESCAPE
                031322  000122                          .WORD   10000$-.
4903
4904            031324                                  BGNSEG                          ;%%START OF SEGMENT%%
                031324  104404                          TRAP    C$BSEG
4905
4906            031326  005037  003426                  CLR     BUF
4907            031332  005077  151022                  CLR     @RLDA
4908            031336  012777  003426  151012          MOV     #BUF,@RLBA      ;SETTING SECTOR 40 OF CYL. ADDR.
4909            031344  012777  177600  151010          MOV     #-128.,@RLMP    ;WORD COUNT
4910
4911            031352  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4912            031356  000002                          WRCHK                   ;WRITE CHECK
4913            031360  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
4914            031364                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
                031364  104410                          TRAP    C$ESCAPE
                031366  000054                          .WORD   10001$-.
4915
4916            031370  013737  002340  002272          MOV     E.CS,TMP0       ;GET RLCS
4917            031376  042737  001777  002272          BIC     #1777,TMP0      ;SAVE ERROR BITS
4918            031404  022737  104000  002272          CMP     #BIT15!BIT11,TMP0  ;DCK SET.
4919            031412  001402                          BEQ     1$              ;YES, CONTINUE
4920            031414  004537  014612                  JSR     R5,CHERR
4921            031420                          1$:     CKLOOP
                031420  104406                          TRAP    C$CLP1
4922
4923            031422  022737  104000  002272          CMP     #BIT15!BIT11,TMP0
4924            031430  001404                          BEQ     2$
4925
4926            031432                                  ERRDF   23.,EM65,ERR0
                031432  104455                          TRAP    C$ERDF
                031434  000027                          .WORD   23
                031436  007364                          .WORD   EM65
                031440  007510                          .WORD   ERR0
4927                                                                    ;WHEN FORCED
4928            031442                          2$:
4929
4930            031442                                  ENDSEG                          ;%%END OF SEGMENT%%
                031442                          10001$:
                031442  104405                          TRAP    C$ESEG
4931            031444                                  ENDSEG                          ;%%END OF SEGMENT%%
                031444                          10000$:
                031444  104405                          TRAP    C$ESEG
4932            031446                          ENDTST                          ;**END OF TEST**
                031446                          L10076:
                031446  104401                          TRAP    C$ETST
4933
4934                                            .SBTTL  **TEST 37** - FORCE DCK WITH WRITE CHECK INTERRUPT
4935
4936            031450                          BGNTST                          ;**START OF TEST**
4937
4938            031450                          STARS
                                                ;;*************************************************************************
4939                                            ;FORCE A DCK IN INTERRUPT MODE
```

TEST 37 - FORCE DCK WITH WRITE CHECK INTERRUPT

```
4940 031450                          STARS
                                     ;;*****************************************************************************
4941
4942 031450  004737  015764                  JSR     PC,HDHOME       ;HEADS OVER TRACK 0
4943 031454                                  CKERFG                  ;HEADS GO HOME OKAY
     031462  104432                          TRAP    C$EXIT
     031464  000322                          .WORD   L10077-.
4944
4945 031466                                  BGNSEG                          ;%%START OF SEGMENT%%
     031466  104404                          TRAP    C$BSEG
4946
4947 031470  012700  003426                  MOV     #BUF,R0         ;SETUP AND WRITE
4948 031474  012701  000200                  MOV     #128.,R1        ;128 WORDS
4949 031500  012720  125252          299$:   MOV     #125252,(R0)+   ;WRITE
4950 031504  005301                          DEC     R1              ;DONE??
4951 031506  001374                          BNE     299$
4952
4953 031510  012777  003426  150640          MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
4954 031516  012777  177600  150636          MOV     #-128.,@RLMP    ;WORD COUNT
4955 031524  005077  150630                  CLR     @RLDA           ;CLEAR DISK ADDRESS
4956 031530  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4957 031534  000012                          WRITE
4958 031536  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
4959 031542                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     031542  104410                          TRAP    C$ESCAPE
     031544  000240                          .WORD   10000$-.
4960 031546  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4961 031552                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     031552  104410                          TRAP    C$ESCAPE
     031554  000230                          .WORD   10000$-.
4962                                 ;VERIFY WRITE WITH READ BEFORE WRCHK
4963
4964 031556  005077  150576                  CLR     @RLDA
4965 031562  012777  003426  150566          MOV     #BUF,@RLBA
4966 031570  012777  177600  150564          MOV     #-128.,@RLMP
4967 031576  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4968 031602  000014                          READ
4969 031604  004537  015700                  JSR     R5,WTCRDY
4970 031610                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     031610  104410                          TRAP    C$ESCAPE
     031612  000172                          .WORD   10000$-.
4971 031614  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
4972 031620                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     031620  104410                          TRAP    C$ESCAPE
     031622  000162                          .WORD   10000$-.
4973
4974 031624                                  BGNSEG                          ;%%START OF SEGMENT%%
     031624  104404                          TRAP    C$BSEG
4975
4976 031626                                  SETPRI  #PRI00
     031626  012700  000000                  MOV     #PRI00,R0
     031632  104441                          TRAP    C$SPRI
4977 031634  005037  002256                  CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE FLAG
4978 031640  005037  003426                  CLR     BUF
4979 031644  005077  150510                  CLR     @RLDA
4980 031650  012777  003426  150500          MOV     #BUF,@RLBA      ;SETTING SECTOR 40 OF CYL. ADDR.
4981 031656  012777  177600  150476          MOV     #-128.,@RLMP    ;WORD COUNT
```

TEST 37 - FORCE DCK WITH WRITE CHECK INTERRUPT

```
4982
4983 031664  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
4984 031670  000102                          WRCHK!INTEN                     ;WRITE CHECK
4985 031672  004537  015700                  JSR     R5,WTCRDY               ;WAIT FOR CONTROLLER READY
4986 031676                                  CKLOOP
     031676  104406                          TRAP    C$CLP1
4987                              ;          SETPRI  #PRIO7                                  ;JSD REV A
4988 031700                                  SETPRI  #PRIO6                                  ;JSD REV A
     031700  012700  000300                  MOV     #PRIO6,R0
     031704  104441                          TRAP    C$SPRI
4989
4990 031706  005737  002256                  TST     INTFLG                  ;DID INTERRUPT OCCUR
4991 031712  001004                          BNE     2$                      ;YES OKAY
4992
4993 031714                                  ERRDF   24.,EM66,ERR0           ;NO INTERRUPT FROM DCK
     031714  104455                          TRAP    C$ERDF
     031716  000030                          .WORD   24
     031720  007421                          .WORD   EM66
     031722  007510                          .WORD   ERR0
4994
4995 031724                          2$:     ESCAPE  SEG                     ;CHECK FOR FL:LOE, ELSE EXIT SEG
     031724  104410                          TRAP    C$ESCAPE
     031726  000054                          .WORD   10001$-.
4996
4997 031730  013737  002340  002272          MOV     E.CS,TMP0               ;GET RLCS
4998 031736  042737  001777  002272          BIC     #1777,TMP0              ;SAVE ERROR BITS
4999 031744  022737  104000  002272          CMP     #BIT15!BIT11,TMP0  ;DCK  SET.
5000 031752  001402                          BEQ     1$                      ;YES, CONTINUE
5001
5002 031754  004537  014612                  JSR     R5,CHERR
5003 031760                          1$:     CKLOOP
     031760  104406                          TRAP    C$CLP1
5004
5005 031762  022737  104000  002272          CMP     #BIT15!BIT11,TMP0
5006 031770  001404                          BEQ     3$
5007 031772                                  ERRDF   25.,EM65,ERR0
     031772  104455                          TRAP    C$ERDF
     031774  000031                          .WORD   25
     031776  007364                          .WORD   EM65
     032000  007510                          .WORD   ERR0
5008                                                         ;WHEN FORCED
5009 032002                          3$:
5010
5011 032002                                  ENDSEG                                  ;**END OF SEGMENT**
     032002                          10001$:
     032002  104405                          TRAP    C$ESEG
5012 032004                                  ENDSEG                                  ;**END OF SEGMENT**
     032004                          10000$:
     032004  104405                          TRAP    C$ESEG
5013 032006                          ENDTST                                  ;**END OF TEST**
     032006                          L10077:
     032006  104401                          TRAP    C$ETST
5014
5015                                 .SBTTL  **TEST 38** - CHECK ZERO FILL ON WRITE WITH WRITE CHECK
5016
5017 032010                          BGNTST                                  ;**START OF TEST**
5018
```

TEST 38 - CHECK ZERO FILL ON WRITE WITH WRITE CHECK

```
5019 032010                            STARS
                                       ;;*****************************************************************
5020                                   ;WHEN WRITING PARTIAL SECTORS (LESS THAN 128 WORDS) THE
5021                                   ;CONTROLLER WILL FILL IN THE REMAINING PORTION OF
5022                                   ;THE SECTOR WITH ZERO WORDS.  CHECK THIS FEATURE CAN BE WRITE CHECKED
5023                                   ;WITH WORD COUNTS FROM 1 TO 127
5024 032010                            STARS
                                       ;;*****************************************************************
5025
5026 032010  004737  015764                    JSR     PC,@DHOME       ;HEADS OVER TRACK 0
5027 032014                                    CKERFG                  ;HEADS GO HOME OKAY
     032022  104432                            TRAP    C$EXIT
     032024  000274                            .WORD   L10100-.
5028
5029 032026                                    BGNSEG                          ;**START OF SEGMENT**
     032026  104404                            TRAP    C$BSEG
5030
5031 032030  012737  000001  002274            MOV     #1,TMP1         ;START WITH 1 WORD WRITE
5032 032036  012700  003426          33$:      MOV     #BUF,R0         ;WRITE BUFFER WITH 52525, WE'LL
5033 032042  012701  000200                    MOV     #128.,R1        ;WRITE 128 WORDS ALL THOUGH WE'RE
5034 032046  012720  052525          3$:       MOV     #52525,(R0)+    ;ONLY GOING TO TRANSFER < 128
5035 032052  005301                            DEC     R1              ;DONE WITH BUFFER?
5036 032054  001374                            BNE     3$              ;NO, GO BACK
5037 032056  013700  002274                    MOV     TMP1,R0         ;GET TRANSFER WORD COUNT
5038 032062  005400                            NEG     R0              ;NEGATE FOR RLMP
5039 032064  010077  150272                    MOV     R0,@RLMP        ;STORE WORD COUNT AWAY
5040 032070  012777  003426  150260            MOV     #BUF,@RLBA      ;SET UP RLBA
5041 032076  005077  150256                    CLR     @RLDA
5042 032102  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
5043 032106  000012                            WRITE                   ;WRITE IT
5044 032110  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR WRITE TO FINISH
5045 032114                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     032114  104410                            TRAP    C$ESCAPE
     032116  000200                            .WORD   10000$-.
5046
5047 032120  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
5048 032124                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     032124  104410                            TRAP    C$ESCAPE
     032126  000170                            .WORD   10000$-.
5049                                   ;VERIFY WRITE WITH READ BEFORE WRCHK
5050
5051 032130  005077  150224                    CLR     @RLDA
5052 032134  012777  003426  150214            MOV     #BUF,@RLBA
5053 032142  013700  002274                    MOV     TMP1,R0
5054 032146  005400                            NEG     R0
5055 032150  010077  150206                    MOV     R0,@RLMP
5056 032154  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
5057 032160  000014                            READ
5058 032162  004537  015700                    JSR     R5,WTCRDY
5059 032166                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     032166  104410                            TRAP    C$ESCAPE
     032170  000126                            .WORD   10000$-.
5060 032172  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
5061 032176                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     032176  104410                            TRAP    C$ESCAPE
     032200  000116                            .WORD   10000$-.
5062
```

```
••TEST 38•• - CHECK ZERO FILL ON WRITE WITH WRITE CHECK

5063 032202                                    BGNSEG                          ;%%START OF SEGMENT%%
     032202  104404                            TRAP    C$BSEG
5064 032204  012777  003426  150144            MOV     #BUF,@RLBA      ;SET UP TO READ
5065 032212  013700  002274                    MOV     TMP1,R0
5066 032216  005400                            NEG     R0
5067 032220  010077  150136                    MOV     R0,@RLMP
5068 032224  005077  150130                    CLR     @RLDA           ;SECTOR
5069 032230  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
5070 032234  000002                            WRCHK
5071 032236  004537  015700                    JSR     R5,WTCRDY       ;WAIT TIL WE FINISH THE WRCHK
5072 032242                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     032242  104410                            TRAP    C$ESCAPE
     032244  000034                            .WORD   10001$-.
5073
5074 032246  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
5075 032252  005737  002236                    TST     T.CRC           ;WAS ERROR A DCK??
5076 032256  001003                            BNE     8$              ;YES, GIVE MOR INFO
5077 032260                            10$:    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     032260  104410                            TRAP    C$ESCAPE
     032262  000016                            .WORD   10001$-.
5078 032264  000405                            BR      99$             ;SKIP AROUND
5079 032266                            8$:     CKLOOP                  ;YES, CHECK FOR LOOP FIRST
     032266  104406                            TRAP    C$CLP1
5080 032270                                    ERRDF   37.,EM64,ERR14
     032270  104455                            TRAP    C$ERDF
     032272  000045                            .WORD   37
     032274  007321                            .WORD   EM64
     032276  010414                            .WORD   ERR14
5081 032300                            99$:                            ;EXIT TEST
5082 032300                                    ENDSEG                          ;%%END OF SEGMENT%%
     032300                            10001$:
     032300  104405                            TRAP    C$ESEG
5083
5084 032302  005237  002274                    INC     TMP1
5085 032306  023727  002274  000200            CMP     TMP1,#128.
5086 032314  001250                            BNE     33$
5087
5088 032316                                    ENDSEG                          ;%%END OF SEGMENT%%
     032316                            10000$:
     032316  104405                            TRAP    C$ESEG
5089 032320                            ENDTST                          ;••END OF TEST••
     032320                            L10100:
     032320  104401                            TRAP    C$ETST
5090
5091                                   .SBTTL  ••TEST 39•• - EXTENDED CHECK OF WRITE CHECK FUNCTION
5092
5093 032322                            BGNTST                          ;••START OF TEST••
5094
5095 032322                            STARS
                                       ;;**************************************************************************
5096                                   ;CHECK OF WRITE CHECK LOGIC UNDER FLAG MODE
5097                                   ;TEST IS DONE WITH ALL BIT PATTERNS
5098                                   ; WE WILL WRITE CHECK A FULL SECTOR (128 WORDS) FROM
5099                                   ;MEMORY (BUF).  WE CHECK THAT NO ERRORS OCCUR.
5100 032322                            STARS
                                       ;;**************************************************************************
5101
```

```
**TEST 39** - EXTENDED CHECK OF WRITE CHECK FUNCTION

 5102 032322  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 5103 032326                                    CKERFG                  ;HEADS GO HOME OKAY
      032334  104432                            TRAP    C$EXIT
      032336  000306                            .WORD   L10101-.
 5104
 5105 032340  022737  000001  002232            CMP     #1,T.DRIVE              ;CHECK TYPE OF DRIVE
 5106 032346  001003                            BNE     22$                     ;NOT RL01 THEN BRANCH
 5107 032350  012703  002670                    MOV     #HDRTAB,R3              ;MOV #HDRTAB TO R3
 5108 032354  000402                            BR      33$                     ;THEN BRANCH
 5109 032356  012703  003050          22$:      MOV     #HTAB,R3                ;MOV #HTAB TO R3 (RL02)
 5110
 5111 032362                          33$:      BGNSEG                  ;START OF SEGMENT
      032362  104404                            TRAP    C$BSEG
 5112
 5113 032364  012700  003426          298$:     MOV     #BUF,R0         ;SETUP AND WRITE
 5114 032370  012701  000200                    MOV     #128.,R1        ;128 WORDS
 5115 032374  011302                            MOV     (R3),R2         ;GET PATTERN
 5116 032376  052702  100000                    BIS     #BIT15,R2
 5117 032402  010220                  299$:     MOV     R2,(R0)+
 5118 032404  005301                            DEC     R1              ;DONE??
 5119 032406  001375                            BNE     299$
 5120
 5121 032410  012777  003426  147740            MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
 5122 032416  012777  177600  147736            MOV     #-128.,@RLMP    ;WORD COUNT
 5123 032424  005077  147730                    CLR     @RLDA           ;CLEAR DISK ADDRESS
 5124 032430  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 5125 032434  000012                            WRITE
 5126 032436  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
 5127 032442                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      032442  104410                            TRAP    C$ESCAPE
      032444  000176                            .WORD   10000$-.
 5128 032446  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 5129 032452                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      032452  104410                            TRAP    C$ESCAPE
      032454  000166                            .WORD   10000$-.
 5130 032456                                    BGNSEG                          ;**START OF SEGMENT**
      032456  104404                            TRAP    C$BSEG
 5131
 5132                                ;VERIFY WRITE WITH READ BEFORE WRCHK
 5133
 5134 032460  005077  147674                    CLR     @RLDA
 5135 032464  012777  003426  147664            MOV     #BUF,@RLBA
 5136 032472  012777  177600  147662            MOV     #-128.,@RLMP
 5137 032500  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 5138 032504  000014                            READ
 5139 032506  004537  015700                    JSR     R5,WTCRDY
 5140 032512                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      032512  104410                            TRAP    C$ESCAPE
      032514  000076                            .WORD   10001$-.
 5141 032516  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 5142 032522                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      032522  104410                            TRAP    C$ESCAPE
      032524  000066                            .WORD   10001$-.
 5143
 5144 032526                                    BGNSEG                          ;**START OF SEGMENT**
      032526  104404                            TRAP    C$BSEG
 5145
```

```
**TEST 39** - EXTENDED CHECK OF WRITE CHECK FUNCTION

5146 032530                                3$:
5147 032530  005077  147624                      CLR     @RLDA
5148 032534  012777  177600  147620              MOV     #-128.,@RLMP    ;WORD COUNT
5149 032542  012777  003426  147606              MOV     #BUF,@RLBA      ;BUS ADDRESS
5150 032550  004537  015054                      JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
5151 032554  000002                             WRCHK                   ;WRITE CHECK
5152
5153 032556  004537  015700                      JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
5154 032562                                      ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     032562  104410                              TRAP    C$ESCAPE
     032564  000024                              .WORD   10002$-.
5155
5156 032566  004537  014612                      JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
5157 032572  005737  002236                      TST     T.CRC
5158 032576  001404                             BEQ     4$
5159
5160 032600                                      ERRHRD  410.,ERR15,EM70
     032600  104456                              TRAP    C$ERHRD
     032602  000632                              .WORD   410
     032604  010462                              .WORD   ERR15
     032606  007472                              .WORD   EM70
5161
5162 032610                                4$:
5163
5164 032610                                      ENDSEG                  ;**END OF SEGMENT**
     032610                                10002$:
     032610  104405                              TRAP    C$ESEG
5165 032612                                      ENDSEG                  ;**END OF SEGMENT**
     032612                                10001$:
     032612  104405                              TRAP    C$ESEG
5166
5167 032614  005723                             TST     (R3)+
5168 032616  022737  000001  002232              CMP     #1,T.DRIVE              ;RL01 OR RL02?
5169 032624  001003                             BNE     60$                     ;RL02? THEN BRANCH
5170 032626  020327  003046                      CMP     R3,#HDREND              ;LAST OF PATERN?
5171 032632  000402                             BR      77$
5172 032634  020327  003234            60$:     CMP     R3,#HEND                ;LAST OF PATTERN (RL02)
5173 032640  001251                     77$:    BNE     298$
5174
5175 032642                                      ENDSEG                  ;**END OF SEGMENT**
     032642                                10000$:
     032642  104405                              TRAP    C$ESEG
5176 032644                                      ENDTST                  ;**END OF TEST**
     032644                                L10101:
     032644  104401                              TRAP    C$ETST
5177                                     .SBTTL  **TEST 40** - READ WITHOUT HEADER COMPARE FUNCTION
5178
5179 032646                                      STARS
                                                 ;;*************************************************************************
5180                                             ;TEST THAT READ WITHOUT HEADER VERIFICATION WORKS.  THIS FUNCTION SHOULD
5181                                             ;READ AT THE NEXT SECTOR ENCOUNTERED.  SET THE RLDA TO 0
5182                                             ;AND ISSUE THE FUNCTION IN FLAG MODE.  UPON COMPLETION CHECK
5183                                             ;FOR ERRORS
5184 032646                                      STARS
                                                 ;;*************************************************************************
5185 032646                                      BGNTST                          ;**START OF TEST**
5186
```

```
**TEST 40** - READ WITHOUT HEADER COMPARE FUNCTION

 5187 032646  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 5188 032652                                    CKERFG                  ;HEADS GO HOME OKAY
      032660  104432                            TRAP    C$EXIT
      032662  000052                            .WORD   L10102-.
 5189
 5190 032664                                    BGNSEG                          ;%%START OF SEGMENT%%
      032664  104404                            TRAP    C$BSEG
 5191
 5192 032666  012777  177600  147466            MOV     #-128.,@RLMP    ;SET UP WORD COUNT
 5193 032674  012777  003426  147454            MOV     #BUF,@RLBA      ;SETUP BUS ADDRESS
 5194 032702  012777  177777  147450            MOV     #-1,@RLDA       ;HEADER SHOULDN'T MATTER
 5195 032710  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 5196 032714  000016                            RDNHD                   ;READ DATA WITHOUT HEADER VERIFY
 5197 032716  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR IT TO FINISH
 5198 032722                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      032722  104410                            TRAP    C$ESCAPE
      032724  000006                            .WORD   10000$-.
 5199
 5200 032726  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 5201
 5202 032732                                    ENDSEG                          ;%%END OF SEGMENT%%
      032732                            10000$:
      032732  104405                            TRAP    C$ESEG
 5203 032734                            ENDTST                          ;**END OF TEST**
      032734                            L10102:
      032734  104401                            TRAP    C$ETST
 5204
 5205                                   .SBTTL  **TEST 41** - READ WITHOUT HEADER COMPARE FUNCTION INTERRUPT
 5206
 5207 032736                            BGNTST                          ;**START OF TEST**
 5208
 5209 032736                            STARS
                                        ;;**************************************************************************
 5210                                   ;TEST THAT READ WITHOUT HEADER VERIFICATION WORKS IN
 5211                                   ;INTERRUPT MODE.
 5212 032736                            STARS
                                        ;;**************************************************************************
 5213
 5214 032736  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 5215 032742                                    CKERFG                  ;HEADS GO HOME OKAY
      032750  104432                            TRAP    C$EXIT
      032752  000114                            .WORD   L10103-.
 5216
 5217 032754                                    BGNSEG                          ;%%START OF SEGMENT%%
      032754  104404                            TRAP    C$BSEG
 5218
 5219 032756  005037  002256                    CLR     INTFLG  ;CLEAR INTERRUPT OCCURANCE FLAG
 5220 032762  012777  177600  147372            MOV     #-128.,@RLMP    ;SET UP WORD COUNT FOR ONE SECTOR
 5221 032770  012777  003426  147360            MOV     #BUF,@RLBA      ;SETUP BUFFER ADDRESS
 5222 032776  012777  177777  147354            MOV     #-1,@RLDA       ;DISK ADDRESS IS A DON'T CARE
 5223 033004                                    SETPRI  #PRI00
      033004  012700  000000                    MOV     #PRI00,R0
      033010  104441                            TRAP    C$SPRI
 5224 033012  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
 5225 033016  000116                            RDNHD!INTEN             ;INTERRUPT ENABLED
 5226 033020  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR INTERRUPT
 5227                                   ;       SETPRI  #PRI07                                  ;JSD REV A
```

```
**TEST 41** - READ WITHOUT HEADER COMPARE FUNCTION INTERRUPT

 5228 033024                            SETPRI  #PRIO6                                  ;JSD REV A
      033024  012700  000300            MOV     #PRIO6,R0
      033030  104441                    TRAP    C$SPRI
 5229 033032                            ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      033032  104410                    TRAP    C$ESCAPE
      033034  000030                    .WORD   10000$-.
 5230
 5231 033036  005737  002256            TST     INTFLG          ;DID IT INTERRUPT
 5232 033042  001004                    BNE     1$              ;IF INTERRUPT GO TO 1$
 5233
 5234 033044                            ERRDF   40.,EM40,ERR0   ;NO INTERRUPT
      033044  104455                    TRAP    C$ERDF
      033046  000050                    .WORD   40
      033050  006321                    .WORD   EM40
      033052  007510                    .WORD   ERR0
 5235 033054                    1$:     ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
      033054  104410                    TRAP    C$ESCAPE
      033056  000006                    .WORD   10000$-.
 5236
 5237 033060  004537  014612            JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
 5238
 5239 033064                            ENDSEG                          ;**END OF SEGMENT**
      033064                    10000$:
      033064  104405                    TRAP    C$ESEG
 5240 033066                    ENDTST                                  ;**END OF TEST**
      033066                    L10103:
      033066  104401                    TRAP    C$ETST
 5241
 5242                           .SBTTL  **TEST 42** - CHECK RD W/O HDR CMP ACTUALLY READS
 5243
 5244 033070                    BGNTST                                  ;**START OF TEST**
 5245
 5246 033070                    STARS
                                ;;**********************************************************************
 5247                           ;CHECK THAT THE READ W/O HDR CMP FUNCTION ACTUALLY READS (INTO MEMORY)
 5248                           ;WE WILL WRITE A PATTERN INTO MEMORY AND THEN ISSUE
 5249                           ;A READ TO OVERLAY THAT PATTERN.  AFTER THE READ
 5250                           ;WE CHECK TO SEE IF THE WRITTEN PATTERN HAS CHANGED.
 5251                           ;IF NOT WE ISSUE IT AGAIN AT THE SAME SECTION AFTER
 5252                           ;HAVING MODIFIED OUR PATTERN IN MEMORY (SINCE THERE IS
 5253                           ;ONE CHANCE THAT THE DISK COULD HAVE OUR PATTERN).  AFTER
 5254                           ;THE SECOND READ WE CHECK THE BUFFER AGAIN.  IF IT'S
 5255                           ;NOT CHANGED WE REPORT AN ERROR
 5256 033070                    STARS
                                ;;**********************************************************************
 5257
 5258 033070  004737  015764            JSR     PC,HDHOME       ;HEADS OVER TRACK 0
 5259 033074                            CKERFG                  ;HEADS GO HOME OKAY
      033102  104432                    TRAP    C$EXIT
      033104  000160                    .WORD   L10104-.
 5260
 5261 033106                            BGNSEG                          ;**START OF SEGMENT**
      033106  104404                    TRAP    C$BSEG
 5262
 5263 033110  012737  024350  002272    MOV     #24350,TMP0     ;SET PATTERN TO WRITE
 5264 033116  005037  002274            CLR     TMP1            ;CLEAR PASS INDICATOR
 5265 033122  012700  003426    1$:     MOV     #BUF,R0         ;SET UP BUFFER BEGINNING
```

TEST 42 - CHECK RD W/O HDR CMP ACTUALLY READS

```
5266 033126  012701  000200            2$:   MOV     #128.,R1
5267 033132  013720  002272                  MOV     TMP0,(R0)+      ;WRITE BUFFER
5268 033136  005301                          DEC     R1              ;DONE??
5269 033140  001374                          BNE     2$              ;NO, GO BACK
5270 033142  012777  000050  147210          MOV     #40.,@RLDA      ;LOAD DISK ADDRESS TO NONSENSE
5271 033150  012777  177600  147204          MOV     #-128.,@RLMP    ;SET WORD COUNT
5272 033156  012777  003426  147172          MOV     #BUF,@RLBA      ;LOAD BUS ADDRESS
5273 033164  012737  003426  002300          MOV     #BUF,@DDAT      ;FOR ERROR PRINT
5274
5275 033172  004537  015054                  JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
5276 033176  000016                          RDNHD                   ;READ W/O HDR CMP
5277 033200  004537  015700                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
5278 033204                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     033204  104410                          TRAP    C$ESCAPE
     033206  000054                          .WORD   10000$-.
5279
5280 033210  004537  014612                  JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
5281 033214                                  ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     033214  104410                          TRAP    C$ESCAPE
     033216  000044                          .WORD   10000$-.
5282
5283 033220  012702  003426            4$:   MOV     #BUF,R2         ;SET TO START COMPARING DATA
5284 033224  022237  002272                  CMP     (R2)+,TMP0      ;DID DATA CHANGE?
5285 033230  001014                          BNE     6$              ;YES, CHECK FOR END
5286                                                                 ;
5287
5288                                                                 ;DATA DIDN'T CHANGE, CHECK
5289 033232  005737  002274                  TST     TMP1            ;IF 1ST OR 2ND TIME?
5290 033236  001005                          BNE     5$              ;2ND-REPORT 1ST-TRY AGAIN
5291
5292 033240  005237  002274                  INC     TMP1            ;INC PASS COUNT
5293 033244  005137  002272                  COM     TMP0            ;COMPLIMENT PATTERN
5294 033250  000724                          BR      1$              ;GODO IT AGAIN
5295
5296 033252                            5$:   ERRDF   20.,EM55,ERR9
     033252  104455                          TRAP    C$ERDF
     033254  000024                          .WORD   20
     033256  006652                          .WORD   EM55
     033260  010102                          .WORD   ERR9
5297
5298 033262                            6$:
5299
5300 033262                                  ENDSEG                          ;**END OF SEGMENT**
     033262                          10000$:
     033262  104405                          TRAP    C$ESEG
5301 033264                          ENDTST                          ;**END OF TEST**
     033264                          L10104:
     033264  104401                          TRAP    C$ETST
5302
5303                                 .SBTTL  **TEST 43** - CHECK RLBA INCREMENT WITH RD W/O HDR CMP
5304
5305 033266                          BGNTST                          ;**START OF TEST**
5306
5307 033266                          STARS
                                     ;;*************************************************************************
5308                                 ;CHECK THAT THE RLBA WILL INCREMENT WITH THE READ W/O HDR CMP
5309                                 ;THE RLBA SHOULD CONTAIN "BUF +256."  AFTER A FULL SECTOR
```

TEST 43 - CHECK RLBA INCREMENT WITH RD W/O HDR CMP

```
5310                                    ;READ.
5311 033266                             STARS
                                        ;;*****************************************************************
5312
5313 033266  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
5314 033272                                    CKERFG                  ;HEADS GO HOME OKAY
     033300  104432                            TRAP    C$EXIT
     033302  000120                            .WORD   L10105-.
5315
5316 033304                                    BGNSEG                          ;%%START OF SEGMENT%%
     033304  104404                            TRAP    C$BSEG
5317
5318 033306  012777  000050  147044            MOV     #40.,@RLDA
5319 033314  012777  003426  147034            MOV     #BUF,@RLBA      ;SET UP BUS ADDRESS
5320 033322  012777  177600  147032            MOV     #-128.,@RLMP    ;WORD COUNT
5321 033330  012737  003426  002300            MOV     #BUF,GDDAT      ;FORM EXPECTED BUS ADDRESS
5322 033336  062737  000400  002300            ADD     #256.,GDDAT     ;AFTER READ
5323
5324 033344  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
5325 033350  000016                            RDNHD                   ;READ W/O HDR CMP
5326 033352  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
5327 033356                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     033356  104410                            TRAP    C$ESCAPE
     033360  000040                            .WORD   10000$-.
5328
5329 033362  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
5330 033366                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     033366  104410                            TRAP    C$ESCAPE
     033370  000030                            .WORD   10000$-.
5331 033372  013737  002342  002302            MOV     E.BA,BDDAT      ;READ 'RLBA' FOR PRESENT ADDRESS
5332 033400  023737  002302  002300            CMP     BDDAT,GDDAT     ;DID 'BA' INCREMENT PROPERLY?
5333 033406  001404                            BEQ     1$              ;YES, CONTINUE
5334
5335 033410                                    ERRDF   21.,EM53,ERR4
     033410  104455                            TRAP    C$ERDF
     033412  000025                            .WORD   21
     033414  006717                            .WORD   EM53
     033416  007654                            .WORD   ERR4
5336
5337 033420                             1$:
5338
5339 033420                                    ENDSEG                          ;%%END OF SEGMENT%%
     033420                             10000$:
     033420  104405                            TRAP    C$ESEG
5340 033422                             ENDTST                          ;**END OF TEST**
     033422                             L10105:
     033422  104401                            TRAP    C$ETST
5341
5342                                    .SBTTL  **TEST 44** - CHECK RLDA DOES INCREMENT WITH RD W/O HDR CMP
5343
5344 033424                             BGNTST                          ;**START OF TEST**
5345
5346 033424                             STARS
                                        ;;*****************************************************************
5347                                    ;CHECK THAT THE RLDA DOES INCREMENT BY ONE AFTER A
5348                                    ;FULL SECTOR READ W/O HDR CMP
5349                                    ;AFTER THE READ THE RLDA SHOULD STILL BE THE INITIAL RLDA + 1
```

TEST 44 - CHECK RLDA DOES INCREMENT WITH RD W/O HDR CMP

```
5350 033424                            STARS
                                       ;;**********************************************************************
5351
5352 033424  004737  015764                    JSR     PC,HDHOME       ;HEADS OVER TRACK 0
5353 033430                                    CKERFG                  ;HEADS GO HOME OKAY
     033436  104432                            TRAP    C$EXIT
     033440  000116                            .WORD   L10106-.
5354
5355 033442                                    BGNSEG                          ;**START OF SEGMENT**
     033442  104404                            TRAP    C$BSEG
5356
5357 033444  012737  000050  002300            MOV     #40.,GDDAT      ;DA TO NONSENSE
5358 033452  013777  002300  146700            MOV     GDDAT,@RLDA     ;SETUP DISK ADDRESS
5359 033460  005237  002300                    INC     GDDAT
5360 033464  012777  177600  146670            MOV     #-128.,@RLMP    ;WORD COUNT
5361 033472  012777  003426  146656            MOV     #BUF,@RLBA      ;SETUP BUS ADDRESS
5362
5363 033500  004537  015054                    JSR     R5,LDFUNC               ;LOAD THE FUNCTION IN NEXT WORD
5364 033504  000016                            RDNHD                   ;READ WITHOUT HEADER COMPARE
5365 033506  004537  015700                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY
5366 033512                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     033512  104410                            TRAP    C$ESCAPE
     033514  000040                            .WORD   10000$-.
5367
5368 033516  004537  014612                    JSR     R5,CHERR                ;CHECK CNTLR FOR ERRORS
5369 033522                                    ESCAPE  SEG             ;CHECK FOR FL:LOE, ELSE EXIT SEG
     033522  104410                            TRAP    C$ESCAPE
     033524  000030                            .WORD   10000$-.
5370
5371 033526  013737  002344  002302            MOV     E.DA,BDDAT      ;READ DISK ADDRESS
5372 033534  023737  002300  002302            CMP     GDDAT,BDDAT     ;DID SECTOR INCREMENT PROPERLY
5373 033542  001404                            BEQ     1$              ;YES. BRANCH  NO. REPORT ERROR
5374
5375 033544                                    ERRDF   22.,EM54,ERR4
     033544  104455                            TRAP    C$ERDF
     033546  000026                            .WORD   22
     033550  006764                            .WORD   EM54
     033552  007654                            .WORD   ERR4
5376
5377 033554                            1$:
5378
5379 033554                                    ENDSEG                          ;**END OF SEGMENT**
     033554                            10000$:
     033554  104405                            TRAP    C$ESEG
5380 033556                            ENDTST                          ;**END OF TEST**
     033556                            L10106:
     033556  104401                            TRAP    C$ETST
5381
5382 033560                            BGNMOD  HRDPRM
5383
5384 033560                                    BGNHRD
     033560  000030                            .WORD L10107-L$HARD/2
5385
5386 033562                                    GPRML   CNTYPE,CNT,1,YES
     033562  005130                            .WORD   T$CODE
     033564  033642                            .WORD   CNTYPE
     033566  000001                            .WORD   1
```

TEST 44 - CHECK RLDA DOES INCREMENT WITH RD W/O HDR CMP

```
5387 033570                                    GPRMA   CSRMSG,CSR,0,160000,177776,YES
     033570  000031                            .WORD   T$CODE
     033572  033647                            .WORD   CSRMSG
     033574  160000                            .WORD   T$LOLIM
     033576  177776                            .WORD   T$HILIM
5388 033600                                    GPRML   DRTYPE,TYPDR,1,YES
     033600  003130                            .WORD   T$CODE
     033602  033674                            .WORD   DRTYPE
     033604  000001                            .WORD   1
5389 033606                                    GPRMA   VECMSG,VECT,0,0,776,YES
     033606  001031                            .WORD   T$CODE
     033610  033716                            .WORD   VECMSG
     033612  000000                            .WORD   T$LOLIM
     033614  000776                            .WORD   T$HILIM
5390 033616                                    GPRMD   BRMSG,PRIOR,0,340,0,7,YES
     033616  002032                            .WORD   T$CODE
     033620  033663                            .WORD   BRMSG
     033622  000340                            .WORD   340
     033624  000000                            .WORD   T$LOLIM
     033626  000007                            .WORD   T$HILIM
5391 033630                                    GPRMD   DRMSG,DRBT,0,03400,0,7,YES
     033630  004032                            .WORD   T$CODE
     033632  033725                            .WORD   DRMSG
     033634  003400                            .WORD   03400
     033636  000000                            .WORD   T$LOLIM
     033640  000007                            .WORD   T$HILIM
5392
5393 033642                                    ENDHRD
                                               .EVEN
     033642                            L10107:
5394
5395 033642     122     114     061   CNTYPE: .ASCIZ  /RL11/
     033645     061     000
5396 033647     102     125     123   CSRMSG: .ASCIZ  /BUS ADDRESS/
     033652     040     101     104
     033655     104     122     105
     033660     123     123     000
5397 033663     102     122     040   BRMSG:  .ASCIZ  /BR LEVEL/
     033666     114     105     126
     033671     105     114     000
5398 033674     104     122     111   DRTYPE: .ASCIZ  /DRIVE TYPE = RL01/
     033677     126     105     040
     033702     124     131     120
     033705     105     040     075
     033710     040     122     114
     033713     060     061     000
5399 033716     126     105     103   VECMSG: .ASCIZ  /VECTOR/
     033721     124     117     122
     033724     000
5400 033725     104     122     111   DRMSG:  .ASCIZ  /DRIVE/
     033730     126     105     000
5401                                           .EVEN
5402
5403 033734                                            ENDMOD
5404
5405 033734                                    BGNMOD  SFTPRM
5406
```

TEST 44 - CHECK RLDA DOES INCREMENT WITH RD W/O HDR CMP

```
5407 033734                                BGNSFT
     033734  000022                        .WORD L10110-L$SOFT/2
5408
5409 033736                                GPRML   DMSG,DLT,1,YES
     033736  000130                        .WORD   T$CODE
     033740  034002                        .WORD   DMSG
     033742  000001                        .WORD   1
5410 033744                                XFERF   1$
     033744  006044                        .WORD   T$CODE
5411 033746                                GPRMD   EMSG,ELT,D,177777,0,177777,YES
     033746  001052                        .WORD   T$CODE
     033750  034107                        .WORD   EMSG
     033752  177777                        .WORD   177777
     033754  000000                        .WORD   T$LOLIM
     033756  177777                        .WORD   T$HILIM
5412 033760                          1$:   GPRML   CMSG,DMPCK,1,YES
     033760  003130                        .WORD   T$CODE
     033762  034026                        .WORD   CMSG
     033764  000001                        .WORD   1
5413 033766                                XFERF   2$
     033766  006044                        .WORD   T$CODE
5414 033770                                GPRMD   LMSG,DLMT,D,177777,1,128.,YES
     033770  004052                        .WORD   T$CODE
     033772  034052                        .WORD   LMSG
     033774  177777                        .WORD   177777
     033776  000001                        .WORD   T$LOLIM
     034000  000200                        .WORD   T$HILIM
5415 034002                          2$:
5416
5417 034002                                ENDSFT
                                           .EVEN
     034002                          L10110:
5418
5419 034002  104  122  117          DMSG:   .ASCIZ  /DROP ON ERROR LIMIT/
     034005  120  040  117
     034010  116  040  105
     034013  122  122  117
     034016  122  040  114
     034021  111  115  111
     034024  124  000
5420 034026  103  117  115          CMSG:   .ASCIZ  /COMPARE DATA ON DCK/
     034031  120  101  122
     034034  105  040  104
     034037  101  124  101
     034042  040  117  116
     034045  040  104  103
     034050  113  000
5421 034052  043  040  117          LMSG:   .ASCIZ  /# OF WORDS IN ERROR REPORTED/
     034055  106  040  127
     034060  117  122  104
     034063  123  040  111
     034066  116  040  105
     034071  122  122  117
     034074  122  040  122
     034077  105  120  117
     034102  122  124  105
     034105  104  000
```

TEST 44 - CHECK RLDA DOES INCREMENT WITH RD W/O HDR CMP

```
 5422 034107     105     122     122  EMSG:   .ASCIZ  /ERROR LIMIT/
      034112     117     122     040
      034115     114     111     115
      034120     111     124     000
 5423
 5424 034123                             ENDMOD
 5425
 5426 034123                         LASTAD
                                         .EVEN
      034124  000000                     .WORD   0
      034126  000000                     .WORD   0
      034130                         L$LAST::
 5427
 5428         000001                  .END
```

SYMBOL TABLE

```
ADR    = 000020 G      CLKINT  014504          C$RDBU= 000007          EM24    005616          ERR9    010102 G
AFREG    003500        CLKTIK  014520          C$REFG= 000047          EM25    005656          EVL    = 000004 G
AFTER    015414        CLNCOD  013514 G        C$RESE= 000033          EM26    005710          E$END  = 002100
ANS    = 000012        CMSG    034026          C$REVI= 000003          EM27    005734          E$LOAD= 000035
ARLBA    003435        CNT    = 000012         C$RFLA= 000021          EM30    006005          E.BA    002342
ARLCS    003430        CNTYPE  033642          C$RPT  = 000025          EM31    006040          E.CS    002340
ARLDA    003443        COMP    003645          C$SEFG= 000046          EM32    006076          E.DA    002344
ARLMP    003451        CONT    013046          C$SPRI= 000041          EM33    006135          E.MP    002346
ASSEMB= 000010         CONTIN  012714          C$SVEC= 000037          EM34    006171          E.MP1   002350
BA16   = 000020        CRDY   = 000200         C$TPRI= 000013          EM35    006227          E.MP2   002352
BA17   = 000040        CRTIM   003521          DAHS   = 000020          EM36    006265          FIFTY   002654
BCCFBK   002266        CSR    = 000000         DATPAT  003236          EM4     004712          FIRST   002312
BCSR     002364        CSRMSG  033647          DCKMES  003626          EM40    006321          FIX     015350
BDDAT    002302        CYLMSK  002314          DEMES   003574          EM41    006361          FNDFNC  002372
BEFORE   015362        C$AU   = 000052         DERFLG  002422          EM42    006417          FRMT1   011052
BEREG    003457        C$AUTO= 000061          DERR   = 040000          EM43    006461          FRMT10  011652
BIT0   = 000001 G      C$BRK  = 000022         DIAGMC= 000000          EM44    006522          FRMT11  012001
BIT00  = 000001 G      C$BSEG= 000004          DLMT   = 000010          EM45    006554          FRMT13  012111
BIT01  = 000002 G      C$BSUB= 000002          DLT    = 000000          EM47    006604          FRMT14  011476
BIT02  = 000004 G      C$CEFG= 000045          DLTMES  003633          EM5     004735          FRMT15  012142
BIT03  = 000010 G      C$CLCK= 000062          DMPCK  = 000006          EM50    006621          FRMT16  012167
BIT04  = 000020 G      C$CLEA= 000012          DMSG    034002          EM53    006717          FRMT17  012233
BIT05  = 000040 G      C$CLOS= 000035          DRBT   = 000010          EM54    006764          FRMT18  012316
BIT06  = 000100 G      C$CLP1= 000006          DRDY   = 000001          EM55    006652          FRMT2   011102
BIT07  = 000200 G      C$CVEC= 000036          DRIVE   002246          EM56    007033          FRMT2A  011121
BIT08  = 000400 G      C$DCLN= 000044          DRMSG   033725          EM57    007052          FRMT2B  011134
BIT09  = 001000 G      C$DODU= 000051          DROP    012434          EM6     004763          FRMT3   011163
BIT1   = 000002 G      C$DRPT= 000024          DRPCOD  013610 G        EM60    007107          FRMT4   011170
BIT10  = 002000 G      C$DU   = 000053         DRST   = 000010          EM61    007137          FRMT5   011226
BIT11  = 004000 G      C$EDIT= 000003          DRTIM   003546          EM62    007207          FRMT6   011277
BIT12  = 010000 G      C$ERDF= 000055          DRTYPE  033674          EM63    007246          FRMT7   011354
BIT13  = 020000 G      C$ERHR= 000056          DSPCOD  012446 G        EM64    007321          FRMT8   011426
BIT14  = 040000 G      C$ERRO= 000060          DS0    = 000000         EM65    007364          FRMT9   011547
BIT15  = 100000 G      C$ERSF= 000054          DS1    = 000400         EM66    007421          FRMT98  012044
BIT2   = 000004 G      C$ERSO= 000057          DS2    = 001000         EM7     005017          FRMT99  012106
BIT3   = 000010 G      C$ESCA= 000010          DS3    = 001400         EM70    007472          F$AU   = 000015
BIT4   = 000020 G      C$ESEG= 000005          DWORD   002322          END     013240          F$AUTO= 000020
BIT5   = 000040 G      C$ESUB= 000003          EF.CON= 000036 G        ERCOUN  002426          F$BGN  = 000040
BIT6   = 000100 G      C$ETST= 000001          EF.NEW= 000035 G        ERFLG   002400          F$CLEA= 000007
BIT7   = 000200 G      C$EXIT= 000032          EF.PWR= 000034 G        ERPOIN  002424          F$DU   = 000016
BIT8   = 000400 G      C$GETB= 000026          EF.RES= 000037 G        ERR    = 100000         F$END  = 000041
BIT9   = 001000 G      C$GETW= 000027          EF.STA= 000040 G        ERRVEC  002244          F$HARD= 000004
BOE    = 000400 G      C$GMAN= 000043          ELT    = 000002         ERR0    007510 G        F$HW   = 000013
BPRIOR   002370        C$GPHR= 000042          EMSG    034107          ERR1    007526 G        F$INIT= 000006
BRMSG    033663        C$GPLO= 000030          EM1     004467          ERR10   010146 G        F$JMP  = 000050
BUF      003426        C$GPRI= 000040          EM10    005064          ERR11   010220 G        F$MOD  = 000000
BVEC     002366        C$INIT= 000011          EM100   004522          ERR12   010272 G        F$MSG  = 000011
B.BA     002332        C$INLP= 000020          EM11    005124          ERR13   010346 G        F$PROT= 000021
B.CS     002330        C$MANI= 000050          EM12    005144          ERR14   010414 G        F$PWR  = 000017
B.DA     002334        C$MEM  = 000031         EM13    005176          ERR15   010462 G        F$RPT  = 000012
B.MP     002336        C$MSG  = 000023         EM14    005224          ERR2    007540 G        F$SEG  = 000003
CALBCC   002270        C$OPEN= 000034          EM16    005300          ERR3    007602 G        F$SOFT= 000005
CDCNT    002242        C$PNTB= 000014          EM17    005322          ERR4    007654 G        F$SRV  = 000010
CHECK    002234        C$PNTF= 000017          EM20    005346          ERR5    007722 G        F$SUB  = 000002
CHERR    014612        C$PNTS= 000016          EM21    005414          ERR6    007760 G        F$SW   = 000014
CKERLT   014526        C$PNTX= 000015          EM22    005461          ERR7    010022 G        F$TEST= 000001
CLKFLD   002666        C$QIO  = 000377         EM23    005540          ERR8    010030 G        GDDAT    002300
```

SYMBOL TABLE

```
GLBDAT  002232 G
GLBEQA  002232 G
GLBERR  007510 G
GLBSUB  013614 G
GLBTXT  003430 G
GODRVR= 000202
GSBIT = 000002
GSTAT = 000004
GSTINT  004221
GSTMES  004171
G$CNTO= 000200
G$DELM= 000372
G$DISP= 000003
G$EXCP= 000400
G$HILI= 000002
G$LOLI= 000001
G$NO  = 000000
G$OFFS= 000400
G$OFSI= 000376
G$PRMA= 000001
G$PRMD= 000002
G$PRML= 000000
G$RADA= 000140
G$RADB= 000000
G$RADD= 000040
G$RADL= 000120
G$RADO= 000020
G$XFER= 000004
G$YES = 000010
HCRCME  003613
HDHOME  015764
HDREND  003046
HDRLST  015310
HDRTAB  002670
HEND    003234
HNFMES  003621
HOE   = 100000 G
HPTCOD  012414 G
HRDPRM  033560 G
HTAB    003050
HZ      002650
IBE   = 010000 G
IDU   = 000040 G
IER   = 020000 G
INITCO  012600 G
INTEN = 000100
INTFLG  002256
INTSRV  014464
ISR   = 000100 G
IXE   = 004000 G
I$AU  = 000041
I$AUTO= 000041
I$CLN = 000041
I$DU  = 000041
I$HRD = 000041
I$INIT= 000041
I$MOD = 000041
I$MSG = 000041
I$PROT= 000040
I$PTAB= 000041
I$PWR = 000041
I$RPT = 000041
I$SEG = 000041
I$SETU= 000041
I$SFT = 000041
I$SRV = 000041
I$SUB = 000041
I$TST = 000041
J$JMP = 000167
LDCSR   002260
LDFUNC  015054
LF      003640
LINE1   010522
LINE2   010556
LINE3   011000
LMSG    034052
LOE   = 040000 G
LOPIMN  002404
LOPIMX  002402
LOT   = 000010 G
L$ACP   002110 G
L$APT   002036 G
L$AUT   002070 G
L$AUTO  013326 G
L$CCP   002106 G
L$CLEA  013514 G
L$CO    002032 G
L$DEPO  002011 G
L$DESC  002122 G
L$DESP  002076 G
L$DEVP  002060 G
L$DISP  012450 G
L$DLY   002116 G
L$DTP   002040 G
L$DTYP  002034 G
L$DU    013610 G
L$DUT   002072 G
L$DVTY  002220 G
L$EF    002052 G
L$ENVI  002044 G
L$ETP   002102 G
L$EXP1  002046 G
L$EXP4  002064 G
L$EXP5  002066 G
L$HARD  033562 G
L$HIME  002120 G
L$HPCP  002016 G
L$HPTP  002022 G
L$HW    012416 G
L$ICP   002104 G
L$INIT  012600 G
L$LADP  002026 G
L$LAST  034130 G
L$LOAD  002100 G
L$LUN   002074 G
L$MREV  002050 G
L$NAME  002000 G
L$PRIO  002042 G
L$PROT  012406 G
L$PRT   002112 G
L$REPP  002062 G
L$REV   002010 G
L$SOFT  033736 G
L$SPC   002056 G
L$SPCP  002020 G
L$SPTP  002024 G
L$STA   002030 G
L$SW    012434 G
L$TEST  002114 G
L$TIML  002014 G
L$UNIT  002012 G
L10000  007524
L10001  007536
L10002  007600
L10003  007652
L10004  007720
L10005  007756
L10006  010020
L10007  010026
L10010  010100
L10011  010144
L10012  010216
L10013  010270
L10014  010344
L10015  010412
L10016  010460
L10017  010520
L10021  012432
L10022  012446
L10023  013324
L10024  013512
L10025  013606
L10026  013612
L10027  014470
L10030  014502
L10031  014516
L10032  014524
L10033  016402
L10034  016532
L10035  016666
L10036  017020
L10037  017156
L10040  017354
L10041  017756
L10042  020146
L10043  020344
L10044  020516
L10045  020714
L10046  021114
L10047  021216
L10050  021342
L10051  021536
L10052  021672
L10053  022024
L10054  022144
L10055  022324
L10056  023136
L10057  023332
L10060  023476
L10061  023662
L10062  024046
L10063  024446
L10064  025070
L10065  025516
L10066  026176
L10067  026630
L10070  027244
L10071  027476
L10072  027766
L10073  030262
L10074  030554
L10075  031146
L10076  031446
L10077  032006
L10100  032320
L10101  032644
L10102  032734
L10103  033066
L10104  033264
L10105  033422
L10106  033556
L10107  033642
L10110  034002
MAXCYL  002324
MAXSEC  002320
MDHEDR  002000 G
MERLMT  012436
MK    = 000001
MSCRLF  003642
MXSEC1  002316
NOOPO = 000000
NOPINT  003733
NOPMES  003705
NOPWR   012626
NOTST   002662
NXM   = 020000
NXMMES  003601
NXT     012724
OPI   = 002000
OPIERR  003653
OPIMES  003606
OPIMN   002412
OPIMX   002414
OPITIM  002664
O$APTS= 000000
O$AU  = 000000
O$BGNR= 000000
O$BGNS= 000001
O$DU  = 000001
O$ERRT= 000000
O$GNSW= 000001
O$POIN= 000001
O$SETU= 000000
PCLKCS  002642
PCLOCK  002660
PCSR    002644
PNT   = 001000 G
PRI   = 002000 G
PRIOR = 000004
PRI00 = 000000 G
PRI01 = 000040 G
PRI02 = 000100 G
PRI03 = 000140 G
PRI04 = 000200 G
PRI05 = 000240 G
PRI06 = 000300 G
PRI07 = 000340 G
PWRFLG  002416
RDDINT  004271
RDDMES  004251
RDHDR = 000010
RDNHD = 000016
RDNINT  004401
RDNMES  004353
READ  = 000014
REST    012772
RESTMS  015036
RHDINT  004103
RHDMES  004053
RHHS  = 000100
RLBA    002356
RLCS    002354
RLDA    002360
RLMP    002362
SECMSK  002262
SEEK  = 000006
SEKINT  004150
SEKMES  004130
SETCLK  014110
SFTPRM  033734 G
SIGN  = 000004
SIMBCC  015462
SIXTY   002656
SIZE  = 000004
SKHOME  004430
SPTCOD  012432 G
START   012732
START1  012646
STHS  = 000100
SVCGBL= 000000
SVCINS= 000000
SVCSUB= 177777
SVCTAG= 000000
SVCTST= 177777
SVHD    002326
```

SYMBOL TABLE

```
S$LSYM= 010000        T$NEST= 177777        T$$SRV= 010032        T27     025072 G      UOPIMN  002410
TAG     002640        T$NS0 = 000000        T$$SW = 010022        T28     025520 G      UOPIMX  002406
TEMP    002634        T$NS1 = 000005        T$$TES= 010106        T29     026200 G      UUT     002250
TEMP0   002632        T$NS2 = 000003        T.ANS   012444        T3      016534 G      VEC     002646
TEMP2   002304        T$NS3 = 000003        T.CNTL  002420        T30     026632 G      VECMSG  033716
TEMP3   002306        T$PTNU= 000000        T.CRC   002236        T31     027246 G      VECT  = 000002
TEMP4   002310        T$SAVL= 177777        T.DMP   012440        T32     027500 G      WCKINT  004016
TIME    013614        T$SEGL= 177777        T.DRIV  002232        T33     027770 G      WCKMES  003762
TIMSRV  014472        T$SEK0= 010000        T.LMT   012442        T34     030264 G      WHY     002240
TIM.US  002636        T$SEK1= 010001        T1      016240 G      T35     030556 G      WRCHK = 000002
TMP0    002272        T$SEK2= 010002        T10     020346 G      T36     031150 G      WRITE = 000012
TMP1    002274        T$SUBN= 000000        T11     020520 G      T37     031450 G      WRLOCK  004452
TMP2    002276        T$TAGL= 177777        T12     020716 G      T38     032010 G      WRTINT  004332
TRPFLG  002254        T$TAGN= 010111        T13     021116 G      T39     032322 G      WRTMES  004311
TRPHAN  015756        T$TEMP= 000000        T14     021220 G      T4      016670 G      WTCRDY  015700
TRYFNC  002376        T$TEST= 000054        T15     021344 G      T40     032646 G      WTDRDY  015634
TYPDR = 000006        T$TSTM= 177777        T16     021540 G      T41     032736 G      XDELAY  002626
T$ARGC= 000004        T$TSTS= 000001        T17     021674 G      T42     033070 G      XITFLG  002652
T$CODE= 004052        T$$AUT= 010024        T18     022026 G      T43     033266 G      XMEM    002374
T$ERRN= 000026        T$$CLE= 010025        T19     022146 G      T44     033424 G      XPOLY   002264
T$EXCP= 000000        T$$DU = 010026        T2      016404 G      T5      017022 G      XTIME   013746
T$FLAG= 000040        T$$HAR= 010107        T20     022326 G      T6      017160 G      XXX     012754
T$GMAN= 000000        T$$HW = 010021        T21     023140 G      T7      017356 G      X$ALWA= 000000
T$HILI= 000200        T$$INI= 010023        T22     023334 G      T8      017760 G      X$FALS= 000040
T$LAST= 000001        T$$MSG= 010017        T23     023500 G      T9      020150 G      X$OFFS= 000400
T$LOLI= 000001        T$$PRO= 010020        T24     023664 G      UAM   = 000200 G      X$TRUE= 000020
T$LSYM= 010000        T$$SEG= 010000        T25     024050 G      UNITST  002252        YDELAY  002630
T$LTNO= 000054        T$$SOF= 010110        T26     024450 G

. ABS.  034130      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  28726 WORDS  ( 113 PAGES)
DYNAMIC MEMORY:  20060 WORDS  ( 77 PAGES)
ELAPSED TIME:  00:36:12
CNRLHA.BIN,CNRLHA.LST/-SP-SVC34.MLB/ML,CNRLHA.MAC
```