RL11,RLV11

CTLR 1
CZRLGA0

AH-F110A-MC

FICHE 1 OF 1

MAY 1979
digital
MADE IN USA

IDENTIFICATION

PRODUCT CODE: AC-F111A-MC

PRODUCT NAME: CZRLGA0 RL11/RLV11 CONTROLLER TEST PART 1

DATE CREATED: 5-JAN-79

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: D. DEKNIS, C. CAMPBELL

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

1.1.1 STRUCTURE OF PROGRAM

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

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

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

1.1.2 DIAGNOSTIC INFORMATION

THE RL11/RLV11 CONTROLLER TEST (PART 1) IS A PDP-11 (LSI-11) BASED PROGRAM THAT WILL TEST THE CONTROLLER. IT STARTS BY TESTING BASIC INTERFACE LOGIC, REGISTER MANIPULATION AND FUNCTIONALITY WHICH INCLUDES NOOP, GET STATUS, READ HEADERS AND SEEK OPERATIONS. IT IS AIMED AT FULLY TESTING THE CONTROLLER IN THESE AREAS, BUT BY DEFAULT ALSO EXERCISES THE DRIVE. THE TEST COVERAGE OF THE PROGRAM IS EXTREMELY HIGH.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

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

SECTOR FILE'
1 - 8 RL02 DRIVES WITH RL02K CARTRIDGES CONTAINING A 'BAD SECTOR FILE'

* KW11P, KW11L (OPTIONAL)

* LINE PRINTER (OPTIONAL)

1.2.2 SOFTWARE REQUIREMENTS

CZRLGA0 RL11/RLV11 CTLR 1

(FORMERLY CZRLAB)

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

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

CVRLAA0 RLV11 RL01 DISKLESS TEST (RLV11 ONLY)

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 SIX STEPS OF EXECUTION

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

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

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

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

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

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

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

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

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

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

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

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

THINGS TO NOTE HERE:

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

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

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

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

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

```
**********
* STEP 3 *
**********
```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

2.1.2 SAMPLE RUN-THROUGH

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

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

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

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

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

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

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

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

THIS WILL DO THE FOLLOWING:

1. TURN ON THE IER (INHIBIT ERROR PRINTOUT) FLAG

2. TURN ON THE LOE FLAG

3. TURN OFF THE HOF FLAG

4. RESUME EXECUTION AT INSTRUCTION AFTER ERROR REPORT

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

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

1. START

2. RESTART

3. CONTINUE

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

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

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

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

1. START

2. RESTART

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

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

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

.R CZRLGA                               O

CZRLG                                   D

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

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

# UNITS (D) ? 2                        D,O

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

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

CHANGE SW (L) ? N                       D,O


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

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


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

```
^C                                          0

DS-B>CON/FLAGS:HOE:1ER:LOE=0                D,0

CHANGE SW (L) ? N                           D,0

CZRLG EOP 1                                 D
^C

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

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

2.2 HOW TO CREATE A CHAINABLE FILE

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

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

HERE IS A SAMPLE DIALOGUE TO ACCOMPLISH THIS:

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

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

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

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

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

```
    .R DIAG.BIC
```

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

2.3 DETAILS OF COMMANDS AND SYNTAX

2.3.1 TABLE OF COMMAND VALIDITY.

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

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

2.3.2 COMMAND SYNTAX

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

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

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

THE SWITCH ARGUMENTS ARE AS FOLLOWS:

"TEST-LIST" IS A SEQUENCE OF DECIMAL NUMBERS (1:2 ETC.) OR RANGES OF DECIMAL NUMBERS (1-5:8-10 ETC.) THAT SPECIFY THE TESTS TO BE EXECUTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS RANGE FROM 1 TO THE LARGEST TEST NUMBER IN THE DIAGNOSTIC. THEY MAY BE SPECIFIED IN ANY ORDER. TESTS WILL BE EXECUTED IN NUMERICAL ORDER REGARDLESS OF THE ORDER OF SPECIFICATION. THE DEFAULT IS TO EXECUTE ALL TESTS.

"PASS-CNT" IS A DECIMAL NUMBER INDICATING THE DESIRED NUMBER OF PASSES. A PASS IS DEFINED AS THE EXECUTION OF THE FULL DIAGNOSTIC (ALL SELECTED TESTS) AGAINST ALL UNITS SUBMITTED. THE DEFAULT IS NON-ENDING TEST EXECUTION. "FLAG-LIST" IS A SEQUENCE OF ELEMENTS OF THE FORM <FLAG>, <FLAG=1>, OR <FLAG=0>, SEPARATED BY COLONS, WHERE <FLAG> HAS ONE OF THE FOLLOWING VALUES:

HOE HALT ON ERROR, CAUSING COMMAND MODE TO BE ENTERED WHEN AN ERROR IS ENCOUNTERED

LOE LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAINING THE ERROR

IER INHIBIT ERROR REPORTING

IBE INHIBIT BASIC ERROR REPORTS

IXE INHIBIT EXTENDED ERROR REPORTS

PRI DIRECT ALL MESSAGES TO A LINE PRINTER

PNT PRINT NUMBER OF TEST BEING EXECUTED

BOE BELL ON ERROR

UAM RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS

ISR INHIBIT STATISTICAL REPORTS

IDU INHIBIT DROPPING OF UNITS BY DIAGNOSTIC

THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED.

"EOP-INCR" IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE PRINTED. THE DEFAULT IS AT THE END OF EVERY PASS.

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

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

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

THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:

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

2. ALL UNSPECIFIED FLAG SETTINGS ARE UNCHANGED.

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

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

THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:

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

2. UNSFECIFIED FLAG SETTINGS ARE UNCHANGED

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

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

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

1. UNSPECIFIED FLAG SETTINGS ARE UNCHANGED

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

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

ALL FLAGS ARE CLEARED.

2.4 EXTENDED P-TABLE DIALOGUE

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

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

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

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

NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO CONSTRUCT A SET OF P-TABLES. ASSUME THAT WE HAVE 8 RL UNITS, AND THAT THERE ARE FIVE (5) HARDWARE PARAMETERS FOR EACH (5 SLOTS IN THE P-TABLE, 5 HARDWARE QUESTIONS IN THE DIALOGUE).

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

```
# UNITS (D) ? 8

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

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

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

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

2.5 HARDWARE PARAMETERS

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

RL11 (L) Y?

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

BUS ADDRESS (O) 174400?

ANSWER WITH THE BUS ADDRESS OF THE CONTROLLER.

VECTOR (O) 160?

ANSWER WITH THE INTERRUPT VECTOR OF THE CONTROLLER.

DRIVE (O) 0?

ANSWER WITH THE DRIVE(S) CONNECTED TO THE CONTROLLER

DRIVE TYPE = RL01 (L) ?

ANSWER NO (N) IF DRIVE IS AN RL02

BR LEVEL (O) 5?

ANSWER WITH THE INTERRUPT PRIORITY OF THE CONTROLLER.

2.6 SOFTWARE PARAMETERS

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

"CHANGE S.W. ?"

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

"DROP ON ERROR LIMIT (L) Y?"

TO ALLOW THE UNIT TO BE DROPPED ONCE A PREDETERMINED NUMBER OF ER-

RORS ARE ENCOUNTERED.

ANSWER Y OR N

"ERROR LIMIT (D) 10?"

NUMBER OF ERRORS ALLOWED BEFORE DROPPING UNIT.

ANSWER 1 TO 65K

"AUTOSIZE (L) N?"

TO CHECK TO SEE IF UNIT SPECIFIED ACTUALLY EXISTS BEFORE TESTING IT (VIA DRIVE READY), IF NOT UNIT WILL NOT BE TESTED.

ANSWER Y OR N

3.0 ERROR INFORMATION

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

3.1 ERROR REPORTING

ALL ERROR INFORMATION IS PRINTED ON THE CONSOLE DECIVE. ERROR REPORTS ARE AIMED AT BEING SELF EXPLANATORY. THE GENERAL FORMAT IS:

```
DZRL? XXX   ERR YYYYY   TST ZZZ   SUB PPP   PC: RRRRRR
```

WHERE:

?	IS PROGRAM LETTER
XXX	IS SFT - SOFT ERROR
	HRD - HARD ERROR
	DV FAT - DEVICE FATAL ERROR
	SYS FAT - SYSTEM FATAL ERROR
YYYYY	IS THE ERROR NUMBER
ZZZ	IS THE TEST NUMBER
PPP	IS THE SUBTEST NUMBER
RRRRRR	IS THE PROGRAM LISTING LOCATION

ERRORS GIVE THE REGISTER CONTENTS BEFORE AND AFTER THE ERROR ALONG WITH A ONE LINE DESCRIPTION AND RELEVENT DATA.

EXAMPLE:

```
ONE LINE DESCRIPTION
(OPTIONAL SECOND LINE)
(OPTIONAL THIRD LINE)
BEFORE COMMAND: CS:XXXXXX BA:XXXXXX DA:XXXXXX MP:XXXXXX
TIME OF ERROR:  CS:XXXXXX BA:XXXXXX DA:XXXXXX MP:XXXXXX XXXXXX
 XXXXXX
```

3.2 ERROR HALTS

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

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS

THIS PROGRAM WILL NOT GIVE ANY PERFORMANCE REPORTS.

4.2 PROGRESS REPORTS

THIS PROGRAM WILL NOT GIVE ANY PROGRESS REPORTS.

5.0 DEVICE INFORMATION TABLES

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

RLCS - CONTROL AND STATUS REGISTER (XXXXX0)

```
BIT 15 - COMPOSITE ERROR
BIT 14 - DRIVE ERROR
BIT 13 - NON EXISTANT MEMORY ERROR
BIT 12 - HEADER NOT FOUND (WITH BIT 10 SET)
       - DATA LATE (WITH BIT 10 CLEAR)
BIT 11 - HEADER CRC (WITH BIT 10 SET)
       - DATA CRC (WITH BIT 10 CLEAR)
BIT 10 - OPERATION INCOMPLETE
BIT  9/8 - DRIVE SELECT (0-3)
BIT  7 - CONTROLLER READY
BIT  6 - INTERRUPT ENABLE
```

```
BIT  5 - EXTENDED BUS ADDRESS (BIT 17)
BIT  4 - EXTENDED BUS ADDRESS (BIT 16)
BIT 3-1 - FUNCTION CODE
        0 - NOP (PDP-11) MAINT (LSI-11)
        1 - WRITE CHECK
        2 - GET DRIVE STATUS
        3 - SEEK
        4 - READ HEADER
        5 - WRITE DATA
        6 - READ DATA
        7 - READ WITHOUT HEADER COMPARE

BIT  0 - DRIVE READY


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

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

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

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

BIT 15-7 - CYLINDER ADDRESS FOR TRANSFER
BIT 6 - SURFACE FOR TRANSFER
BIT 5-0 - SECTOR FOR TRANSFER (1-40.)

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

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

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

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

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

FOR READ/WRITE FUNCTION

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

FOR READ HEADER FUNCTION

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

FOR GET STATUS FUNCTION

HAS DRIVE STATUS

BIT 15 - WRITE DATA ERROR
BIT 14 - CURRENT HEAD ERROR (CHE)
BIT 13 - WRITE LOCK STATUS (WL)
BIT 12 - SEEK TIME OUT (SKTO)
BIT 11 - SPIN ERROR (SPE)
BIT 10 - WRITE GATE ERROR (WGE)
BIT 9 - VOLUME CHECK (VC)
BIT 8 - DRIVE SELECT ERROR (DSE)
BIT 7 - DRIVE TYPE IS RL02 IF SET
BIT 6 - SURFACE (0=UPPPER, 1=LOWER)
BIT 5 - COVER OPEN
BIT 4 - HEADS HOME
BIT 3 - BRUSHES HOME
BIT 2-0 -STATE BITS
0 - LOAD STATE
1 - SPIN UP
2 - BRUSH CYCLE
3 - LOAD HEADS
4 - SEEK - TRACK COUNTING
5 - SEEK - LINEAR MODE
6 - UNLOAD HEADS
7 - SPIN DOWN

6.0 TEST SUMMARIES

TEST 1 - RLCS ADDRESSABILITY

THIS TEST WILL CHECK THAT THE CONTROL AND STATUS REGISTER CAN BE ACCESSED. IF THE REGISTER CANNOT BE ACCESSED THE PROCESSOR WILL TRAP TO LOCATION 4, WHICH IS SET UP TO HANDLE THE TRAP.

TEST 2 - RLBA ADDRESSABILITY
******

THIS TEST WILL CHECK THAT THE BUS ADDRESS REGISTER CAN BE ACCESSED. IF THE REGISTER CANNOT BE ACCESSED THE PROCESSOR WILL TRAP TO LOCATION 4, WHICH IS SET UP TO HANDLE THE TRAP.

TEST 3 - RLDA ADDRESSABILITY
******

THIS TEST WILL CHECK THAT THE DISK ADDRESS REGISTER CAN BE ACCESSED. IF THE REGISTER CANNOT BE ACCESSED THE PROCESSOR WILL TRAP TO LOCATION 4, WHICH IS SET UP TO HANDLE THE TRAP.

TEST 4 - RLMP ADDRESSABILITY
******

THIS TEST WILL CHECK THAT THE MULTIPURPOSE REGISTER CAN BE ACCESSED. IF THE REGISTER CANNOT BE ACCESSED THE PROCESSOR WILL TRAP TO LOCATION 4, WHICH IS SET UP TO HANDLE THE TRAP.

TEST 5 - READ WRITE OF RLCS
******

THIS TEST WILL ATTEMPT TO WRITE RLCS BITS 9-1 AND READ THEM BACK. WALKING AND GROWING 0'S AND 1'S ARE USED. BIT 7 (CONTROLLER READY) IS ALWAYS WRITTEN AS A 1 SO NOT TO INITIATE A FUNCTION. BITS 15, 14 AND 0 ARE TREATED AS DON'T CARE FOR THIS TEST.

TEST 6 - READ WRITE OF RLBA
******

THIS TEST WILL ATTEMPT TO WRITE RLBA BITS 15-0 AND READ THEM BACK. WALKING AND GROWING 0'S AND 1'S ARE USED. BIT 0 ON A RL11 SHOULD ALWAYS COME BACK AS A 0, WHILE ON AN RLV11 IT IS LOADABLE.

TEST 7 - READ WRITE OF RLDA
******

THIS TEST WILL ATTEMPT TO WRITE RLDA BITS 15-0 AND READ THEM BACK. WALKING AND GROWING 0'S AND 1'S ARE USED.

TEST 8 - BIS OF RLCS
******

THIS TEST WILL USE THE 11 INSTRUCTION "BIS" TO SHOW THAT A READ-MODIFY-WRITE SEQUENCE OF THE RLCS WORKS. BITS 9-1 ARE USED, BIT SETTING IN WALKING AND GROWING 0'S AND 1'S. BIT 7 (CONTROLLER READY) IS ALWAYS SET. BITS 15, 14 AND 1 ARE DON'T CARES.

TEST 9 - BIC OF RLCS
******

THIS TEST WILL USE THE 11 INSTRUCTION "BIC" TO SHOW THAT A READ-MODIFY-WRITE SEQUENCE OF THE RLCS WORKS. BITS 9-1 ARE USED, BIT CLEARING IN WALKING AND GROWING 0'S AND 1'S. BIT 7 (CONTROLLER READY) IS ALWAYS SET. BITS 15, 14 AND 1 ARE DON'T CARES.

TEST 10 - BIS OF RLBA
****

THIS TEST WILL USE THE 11 INSTRUCTION "BIS" TO SHOW THAT A READ-MODIFY-WRITE SEQUENCE OF THE RLBA WORKS. BITS 15-0 ARE BIT SET USING GROWING AND WALKING 0'S AND 1'S. BIT 0 CANNOT SET ON A RL11, BUT CAN ON A RLV11.

TEST 11 - BIC OF RLBA
****

THIS TEST WILL USE THE 11 INSTRUCTION "BIC" TO SHOW THAT A READ-MODIFY-WRITE SEQUENCE OF THE RLBA WORKS. BITS 15-0 ARE BIT CLEARED USING GROWING AND WALKING 0'S AND 1'S.

TEST 12 - BIS OF RLDA
****

THIS TEST WILL USE THE 11 INSTRUCTION "BIS" TO SHOW THAT A READ-MODIFY-WRITE SEQUENCE OF THE RLDA WORKS. BITS 15-0 ARE BIT SET USING GROWING AND WALKING 0'S AND 1'S.

TEST 13 - BIC OF RLDA
****

THIS TEST WILL USE THE 11 INSTRUCTION "BIC" TO SHOW THAT A READ-MODIFY-WRITE SEQUENCE OF THE RLDA WORKS. BITS 15-0 ARE BIT CLEARED USING GROWING AND WALKING 0'S AND 1'S.

TEST 14 - BUS RESET OF RLCS
****

THIS TEST WILL VERIFY THAT THE BUS RESET OF THE PROCESSOR WILL CLEAR ALL BITS OF THE RLCS WITH THE EXCEPTION OF BIT 7 (CONTROLLER READY), BIT 0 (DRIVE READY) AND BIT 15 (COMPOSITE ERROR) IF BIT 14 (DRIVE ERROR) IS SET.

TEST 15 - BUS RESET OF RLBA
****

THIS TEST WILL VERIFY TAHT THE BUS RESET OF THE PROCESSOR WILL CLEAR ALL BITS OF THE RLBA.

TEST 16 - BUS RESET OF RLDA
****

THIS TEST WILL VERIFY THAT THE BUS RESET OF THE PROCESSOR WILL CLEAR ALL BITS OF THE RLDA.

TEST 17 - UNIQUENESS OF RLCS
****

THIS TEST WILL VERIFY THAT WHEN THE RLCS (XXXXX0) IS ADDRESSED ONLY THAT REGISTER IS EFFECTED. BOTH THE RLBA AND THE RLDA ARE SET UP WITH KNOWN DATA, THE RLDA IS WRITTEN, THEN THE RLBA AND RLDA ARE VERIFIED THAT THEY DID NOT CHANGE.

TEST 18 - UNIQUENESS OF RLBA
****

THIS TEST WILL VERIFY THAT WHEN THE RLBA (XXXXX2) IS ADDRESSED ONLY THAT REGISTER IS EFFECTED. BOTH THE RLCS AND RLDA ARE WRITTEN WITH KNOWN DATA, THE RLBA IS WRITTEN, THEN THE RLCS AND RLDA ARE VERIFIED THAT THEY DID NOT CHANGE.

TEST 19 - UNIQUENESS OF RLDA
****

THIS TEST WILL VERIFY THAT WHEN THE RLDA (XXXXX4) IS ADDRESSED ONLY THAT REGISTER IS EFFECTED. BOTH THE RLCS AND RLBA ARE WRITTEN WIEHT KNOWN DATA, THE RLDA IS WRITTEN, THEN THE RLCS AND RLBA ARE VERIFIED THAT THEY DID NOT CHANGE.

TEST 20 - UNIQUENESS OF RLMP
****

THIS TEST WILL VERIFY THAT WHEN THE RLMP (XXXXX6) IS ADDRESSED ONLY THAT REGISTER IS EFFECTED. THE RLCS, RLBA AND RLDA ARE

WRITTEN WITH KNOW DATA, THE RLMP IS WRITTEN, THEN THE RLCS, RLBA AND RLDA ARE VERIFIED THAT THEY DID NOT CHANGE.

TEST 21 - NOOP FUNCTION
****

THIS TEST WILL VERIFY THE OPERATION OF THE NOOP (0) FUNCTION ON PDP-11'S ONLY, SINCE ON AN LSI-11 IT IS A MAINTENANCE FUNCTION. THE ABILITY OF CONTROLLER READY TO RESET AND NO ERRORS ARE CHECKED.

TEST 22 - TEST NOOP DOES NOTHING
****

THIS TEST WILL CHECK THAT THE NOOP FUNCTION WILL NOT DISTRUB ANY REGISTERS OF THE CONTROLLER.

TEST 23 - TEST OF INTERRUPT
****

THIS TEST WILL CAUSE AN INTERRUPT FROM THE CONTROLLER USING NOOP (RL11 ONLY) TO CHECK THE INTERRUPT LOGIC AND VECTOR.

TEST 24 - TEST PRIORITY BR LEVEL
****

THIS TEST WILL CHECK THAT THE PROPER PRIORITY IS ON THE BOARD. WE VERIFY THAT ABOVE THE LEVEL THE BOARD WILL NOT INTERRUPT AND BELOW IT, IT WILL.

TEST 25 - GET STATUS FUNCTION
****

THIS TEST WILL VERIFY THAT THE GET STATUS FUNCTION (2) WILL COMPLETE CORRECTLY. THE RLDA IS SET UP AND GET STATUS IS ISSUED. CONTROLLER READY IS CHECKED AS WELL AS ERROR BITS. (FIRST TEST A DRIVE MUST BE PRESENT.)

TEST 26 - GET STATUS FUNCTION INTERRUPT
****

THIS TEST WILL VERIFY THAT THE GET STATUS FUNCTION WILL GENERATE AN INTERRUPT ON COMPLETION.

TEST 27 - GET STATUS FUNCTION GENERATES OPI
****

THIS TEST WILL PROVE THE ABILITY FOR OPI (OPERATION INCOM) TO SET AND THAT THE DRIVE COMMAND IS BEING TRANSMITTED CORRECTLY. THE COMMAND WORD (RLDA) IS SET UP WITH THE MARKER BIT ONLY. AN OPI IS EXPECTED TO RESULT, THIS IS CHECKED.

TEST 28 - OPI UNDER INTERRUPT
****

THIS TEST WILL CHECK THE ABILITY OF AN OPI TO CAUSE AN INTERRUPT TO OCCUR. WE SEND ONLY THE MARKER BIT WITH THE GET STATUS COMMAND AND EXPECT AN OPI ERROR.

TEST 30 - READ HEADER FUNCTION INTERRUPT
****

THIS TEST WILL CHECK THE ABILITY OF THE READ HEADER FUNCTION TO INTERRUPT ON COMPLETION.

TEST 31 - REPEATED RD HDRS YIELD SAME CYL AND HD
****

THIS TEST WILL CHECK THAT ON REPEATED READ HEADERS THE CYLINDER AND HEAD BITS OF THE HEADER WORD (RLMP) ARE ALWAYS THE SAME.

TEST 32 - CHECK OF HEADER CRC
****

THIS TEST WILL VERIFY THE HEADER CRC THAT FOLLOWS THE TWO HEADER WORDS IS ACTUALLY THE CORRECT CRC-16 CALCULATION OF THE TWO HEADER WORDS.

TEST 33 - CHECK CONSECUTIVE HEADERS
****

THIS TEST WILL CHECK THAT HEADERS ARE CONSECUTIVE.

TEST 34 - SEEK FUNCTION
****

THIS TEST WILL CHECK THE SEEK FUNCTION (3) TO RESET CONTROLLER READY AND POST NO ERRORS. COMMAND WORD IS LOADED WITH A ONE CYLINDER FORWARD SEEK.

TEST 35 - CHECK DRIVE READY ON SEEK
****

THIS TEST WILL CHECK THAT DRIVE READY CLEARS AND RESETS ON ISSUEANCE OF A SEEK COMMAND.

TEST 36 - SEEK FUNCTION INTERRUPT
****

THIS TEST WILL CHECK THE ABILITY OF A SEEK COMMAND TO GENERATE AN INTERRUPT ON CONTROLLER READY RESETTING AND NOT ONE ON DRIVE READY RESETTING.

TEST 37 - TEST DIFFERENCE WORD TRANSMISSION
****

THIS TEST WILL TRY TO VERIFY THAT BITS 14-7, 6, 2, 0 OF THE COMMAND WORD GET TRANSMITTED CORRECTLY. WE ISSUE SEEKS FROM TRACK 0 WITH COMMAND WORDS OF WALKING AND GROWING 0'S AND 1'S. ALL SEEKS ARE VERIFIED WITH A READ HEADER AND RETURN TO TRACK 0 BEFORE NEXT PATTERN IS ISSUED.

TEST 38 - VERIFY HEAD SELECT 0 VIA RD HEADER
****

THIS TEST WILL VERIFY THAT HEAD 0 CAN BE SELECTED AND READ VIA READ HEADER.

TEST 39 - VERIFY HEAD SELECT 1 VIA RD HEADER
****

THIS TEST WILL VERIFY THAT HEAD 1 CAN BE SELECTED AND READ VIA READ HEADER.

TEST 40 - VERIFY HEAD SELECT 0 VIA GET STATUS
****

THIS TEST WILL VERIFY THE WORD RETURNED TO THE RLMP BY A GET STATUS CONTAINS THE RIGHT HEAD SELECT.

TEST 41 - VERIFY HEAD SELECT 1 VIA GET STATUS
****

THIS TEST WILL VERIFY THE WORD RETURNED TO THE RLMP BY A GET STATUS CONTAINS THE RIGHT HEAD SELECT.

TEST 42 - TEST TIME AT WHICH DP WD GETS
****

THIS TEST WILL CHECK THAT THE DIFFERENCE WORD (RLDA) ACTUALLY
DOES GET TRANSMITTED PRIOR TO CONTROLLER READY RESETTING.
THIS IS DONE BY ISSUING A SEEK, WAITING FOR CONTROLLER READY
AND RE-LOADING THE RLDA. THE SEEK IS THEN VERIFIED TO SEE IF
IT IS CORRECT.

TEST 43 - EXTENSIVE CHECK OF CRC
****

THIS TEST WILL MORE EXTENSIVELY CHECK THE CRC LOGIC BY POSI-
TIONING AT DIFFERENT POINTS ON THE PACK AND CHECKING THAT THE
HEADER CRC RECEIVED IS CORRECT.

TEST 44 - VERIFY GET STATUS WHILE DRDY IS LOW
****

THIS TEST WILL CHECK THE ABILITY TO PERFORM A GET STATUS WHILE
THE DRIVE IS SEEKING.

TABLE OF CONTENTS

```
                                .TITLE  CZRLGA0 RL11/RLV11 CTLR 1
                                .ENABLE AMA
000000                          .ENABLE ABS
                                .NLIST  ME,CND,MD
                                .MCALL SVC

000000                          SVC
        000000                  SVCINS=0
        000000                  SVCTAG=0
        002000                  .=2000

002000                                  POINTER BGNSFT,BGNSW,BGNDU,BGNAU

002000                          BGNMOD  MDHEDR

002000                                  HEADER  CZRLG,A,0,60,60,4,RL01
002000      103                         .ASCII /C/
002001      132                         .ASCII /Z/
002002      122                         .ASCII /R/
002003      114                         .ASCII /L/
002004      107                         .ASCII /G/
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  000004                          .WORD   4
002016  023734                          .WORD   L$HARD
002020  024110                          .WORD   L$SOFT
002022  011420                          .WORD   L$HW
002024  011436                          .WORD   L$SW
002026  024514                          .WORD   L$LAST
002030  000000                          .WORD   0
002032  000000                          .WORD   0
002034  000000                          .WORD   0
002036  000000                          .WORD   0
002040  011446                          .WORD   L$DISPATCH
002042  000000                          .WORD   0
002044  000000                          .WORD   0
002046  000000                          .WORD   0
002050      002                         .BYTE   C$REVISION
002051      002                         .BYTE   C$EDIT
002052  000060                          .WORD   60
002054  000060                          .WORD   60
002056  000000                          .WORD   0
002060  000000                          .WORD   0
002062  000000                          .WORD   0
002064  002114                          .WORD   L$DVTYP
002066  000000                          .WORD   0
002070  002112                          .WORD   L$DR
002072  002112                          .WORD   L$DRST
002074  012512                          .WORD   L$AU
002076  012506                          .WORD   L$DU
002100  000014                          .WORD   14
002102  000000                          .WORD   0
002104  011576                          .WORD   L$INIT
```

```
      002106  012440                          .WORD   L$CLEAN
   17
   18 002110                                  ENDMOD
   19
   20 002110                                  DEVREG
      002110  000000                          .WORD   0
                                              .BLKW
   21
   22 002114                                  DEVTYP  <RL01,RL02>
      002114      122     114     060         .ASCIZ  /RL01,RL02/
      002117      061     054     122
      002122      114     060     062
      002125      000
                                              .EVEN
   23 002126                          BGNMOD  GLBEQAT
   24
   25 002126                                  EQUALS
   26            000001                       DRDY=BIT0                ;DRIVE READY (RLCS)
   27            000100                       INTEN=BIT6               ;INTERRUPT ENABLE (RLCS)
   28            100000                       ERR=BIT15                ;RL11 ERROR (RLCS)
   29            040000                       DERR=BIT14               ;RL01 DRIVE ERROR (RLCS)
   30            002000                       OPI=BIT10                ;OPERATION INCOMPLETE (RLCS)
   31            000200                       CRDY=BIT7                ;CONTROLLER READY (RLCS)
   32            000040                       BA17=BIT5                ;EXTENDED ADDRESS BIT 17 (RLCS)
   33            000020                       BA16=BIT4                ;EXTENDED ADDRESS BIT 16 (RLCS)
   34            020000                       NXM=BIT13                ;NON-EXISTANT MEMORY (RLCS)
   35            000000                       DS0=0                    ;DRIVE SELECT 0 (RLCS)
   36            000400                       DS1=BIT8                 ;DRIVE SELECT 1 (RLCS)
   37            001000                       DS2=BIT9                 ;DRIVE SELECT 2 (RLCS)
   38            001400                       DS3=BIT8!BIT9            ;DRIVE SELECT 3 (RLCS)
   39            000000                       NOOP0=0                  ;FUNCTION-NOOP(0)
   40            000016                       NOOP7=BIT1!BIT2!BIT3     ;FUNCTION-NOOP(7)
   41            000002                       WRCHK=BIT1               ;WRITE CHECK FUNCTION
   42            000004                       GSTAT=BIT2               ;GET STATUS FUNCTION
   43            000006                       SEEK=BIT2!BIT1           ;SEEK FUNCTION
   44            000010                       RDHDR=BIT3               ;READ HEADER FUNCTION
   45            000012                       WRITE=BIT3!BIT1          ;WRITE DATA FUNCTION
   46            000014                       READ=BIT3!BIT2           ;READ DATA FUNCTION
   47            000202                       GODRVR=BIT1!BIT7         ;CRDY AND DRDY
   48            000010                       DRST=BIT3                ;DRIVE RESET (RLDA)
   49            000002                       GSBIT=BIT1               ;GET STATUS BIT (RLDA)
   50            000001                       MK=BIT0                  ;MARKER BIT (RLDA)
   51            000004                       SIGN=BIT2                ;SIGN BIT (RLDA)
   52            000100                       RHHS=BIT6                ;HEAD SELECT IN READ HEADER
   53            000100                       STHS=BIT6                ;HEAD SELECT IN STATUS BACK
   54            000020                       DAHS=BIT4                ;HEAD SELECT IN SEEK
   55
   56                                         ;OFFSET FOR HARDWARE P-TABLE
   57
   58            000000                       CSR=0
   59            000002                       VECT=2
   60            000004                       PRIOR=4
   61            000006                       TYPDR=6
   62            000010                       DRBT=10
   63            000012                       CNT=12
   64
   65                                         ;OFFSET FOR SOFTWARE P-TABLE
```

```
                        DLT=0
                        ELT=2
                        SIZE=4

002126                          ENDMOD

002126                  BGNMOD  GLBDAT

                        .SBTTL  GLOBAL DATA

002126  000000          PWRFLG: .WORD   0
002130  000000          UUT:    .WORD   0
002132  000000          UNITST: .WORD   0
002134  000000          RLCS:   .WORD   0
002136  000000          RLBA:   .WORD   0
002140  000000          RLDA:   .WORD   0
002142  000000          RLMP:   .WORD   0
002144  000000          BCSR:   .WORD   0
002146  000000          BPRIOR: .WORD   0
002150  000000          BVEC:   .WORD   0
002152  000000          DRIVE:  .WORD   0               ;DRIVE UNDER TEST
002154  000000          B.CS:   .WORD   0
002156  000000          B.BA:   .WORD   0
002160  000000          B.DA:   .WORD   0
002162  000000          B.MP:   .WORD   0
002164  000000          DERFLG: .WORD
002166  000000          E.CS:   .WORD   0
002170  000000          E.BA:   .WORD   0
002172  000000          E.DA:   .WORD   0
002174  000000          E.MP:   .WORD   0
002176  000000          E.MP1:  .WORD   0
002200  000000          E.MP2:  .WORD   0
002202  000000          PFLG:   .WORD   0               ;PROCESSOR TYPE  0=UNIBUS  1=Q-BUS
002204  000000          TRPFLG: .WORD   0
002206  000000          INTFLG: .WORD   0               ;INTERRUPT OCCURANCE FLAG
002210  000000          LDCSR:  .WORD   0               ;LOCATION TO FORM RLCS
002212  000077          SECMSK: .WORD   77              ;MASK OUT SECTOR
002214  120001          XPOLY:  .WORD   120001
002216  000004          ERRVEC: .WORD   4
002220  000000          BCCFBK: .WORD   0               ;LOCATION USED BY "SIMBCC"
002222  000000          CALBCC: .WORD   0               ;LOCATION USED BY "SIMBCC"
002224  000000          TEMP2:  .WORD   0               ;LOCATION USED BY "SIMBCC"
002226  000000          TEMP3:  .WORD   0               ;LOCATION USED BY "SIMBCC"
002230  000000          TEMP4:  .WORD   0               ;LOCATION USED BY "SIMBCC"
002232  000000          TMP0:   .WORD   0
002234  000000          TMP1:   .WORD   0
002236  000000          TMP2:   .WORD   0
002240  000000          GDDAT:  .WORD   0
002242  000000          BDDAT:  .WORD   0
002244  000000          FIRST:  .WORD   0               ;FIRST SECTOR READ
002246  177700          CYLMSK: .WORD   177700          ;MASK CYLINDER AND HEAD SELECT
002250  000050          MXSEC1: .WORD   40.             ;MAX SECTOR ADDRESS +1
002252  000047          MAXSEC: .WORD   39.             ;MAX SECTOR ADDRESS
002254  000000          DWORD:  .WORD   0               ;DIFFERENCE WORD (SEEK)
002256  177600          MAXCYL: .WORD   177600          ;MAXIMUM CYLINDER ADDRESS
002260  000000          SVHD:   .WORD   0               ;SAVE CURRENT HEAD SELECT
```

```
                                WHY:    .WORD   0               ;REASON FOR DROP IN AUTOSIZE
                                T.DRIVE:        .WORD   0
                                T.CNTLR:        .WORD   0
                                TMPFNC: .WORD   0

                                .SBTTL  PATTERNS FOR REGISTER R/W
                                ;
                                ;PATTERNS USED FOR LOADING/READING REGISTERS

002272  000000                  BEGPAT: 0                       ;GROWING 1
002274  000001                          1
002276  000003                          3
002300  000007                          7
002302  000017                          17
002304  000037                          37
002306  000077                          77
002310  000177                          177
002312  000377                          377
002314  000777                          777
002316  001777                          1777
002320  003777                          3777
002322  007777                          7777
002324  017777                          17777
002326  037777                          37777
002330  077777                          77777
002332  177777                          177777
002334  177776                          177776          ;GROWING 0
002336  177774                          177774
002340  177770                          177770
002342  177760                          177760
002344  177740                          177740
002346  177700                          177700
002350  177600                          177600
002352  177400                          177400
002354  177000                          177000
002356  176000                          176000
002360  174000                          174000
002362  170000                          170000
002364  160000                          160000
002366  140000                          140000
002370  100000                          100000

002372  000000                          000000
002374  000001                          1               ;WALKING 1
002376  000002                          2
002400  000004                          4
002402  000010                          10
002404  000020                          20
002406  000040                          40
002410  000100                          100
002412  000200                          200
002414  000400                          400
002416  001000                          1000
002420  002000                          2000
002422  004000                          4000
002424  010000                          10000
```

```
     180 002426  020000                          20000
     181 002430  040000                          40000
     182 002432  100000                          100000
     183 002434  177777                          177777          ;WALKING 0
     184 002436  177776                          177776
     185 002440  177775                          177775
     186 002442  177773                          177773
     187 002444  177767                          177767
     188 002446  177757                          177757
     189 002450  177737                          177737
     190 002452  177677                          177677
     191 002454  177577                          177577
     192 002456  177377                          177377
     193 002460  176777                          176777
     194 002462  175777                          175777
     195 002464  173777                          173777
     196 002466  167777                          167777
     197 002470  157777                          157777
     198 002472  137777                          137777
     199 002474  077777                          077777
     200 002476  177777                          177777
     201 002500  000000                ENDPAT:   000000
     202
     203                                .SBTTL  PATTERNS FOR DIFFERENCE WORD
     204
     205 002502  000200                SKLST:    .WORD   BIT7
     206 002504  000400                          .WORD   BIT8          ;SHIFTING 1
     207 002506  001000                          .WORD   BIT9
     208 002510  002000                          .WORD   BIT10
     209 002512  004000                          .WORD   BIT11
     210 002514  010000                          .WORD   BIT12
     211 002516  020000                          .WORD   BIT13
     212 002520  040000                          .WORD   BIT14
     213 002522  077600                          .WORD   77600         ;SHIFTING 0
     214 002524  077400                          .WORD   77400
     215 002526  076600                          .WORD   76600
     216 002530  075600                          .WORD   75600
     217 002532  073600                          .WORD   73600
     218 002534  067600                          .WORD   67600
     219 002536  057600                          .WORD   57600
     220 002540  037600                          .WORD   37600
     221 002542  077600                          .WORD   77600
     222 002544  000200                          .WORD   200
     223 002546  000600                          .WORD   600           ;GROWING 1
     224 002550  001600                          .WORD   1600
     225 002552  003600                          .WORD   3600
     226 002554  007600                          .WORD   7600
     227 002556  017600                QUAMAX:   .WORD   17600
     228 002560  037600                HALMAX:   .WORD   37600
     229 002562  077600                          .WORD   77600
     230 002564  077400                          .WORD   77400         ;GROWING 0
     231 002566  077000                          .WORD   77000
     232 002570  076000                          .WORD   76000
     233 002572  074000                          .WORD   74000
     234 002574  070000                          .WORD   70000
     235 002576  060000                          .WORD   60000
     236 002600  040000                          .WORD   40000
```

```
                                SKEND:  .WORD   00000
                                RL2:    .WORD   BIT15
                                QMAX:   .WORD   37600
                                HMAX:   .WORD   77600

                                        .WORD   177600
                                        .WORD   177400
                                        .WORD   176600
                                        .WORD   173600
                                        .WORD   167600
                                        .WORD   157600
                                        .WORD   137600
                                        .WORD   177000
                                        .WORD   176000
                                        .WORD   174000
                                        .WORD   170000
                                        .WORD   60000
                                        .WORD   40000
                                SKEEND: .WORD   000000

                                ;PATTERNS FOR TEST OF RLCS

                                CSPAT:  .WORD   0                ;SHIFTING 1
                                        .WORD   BIT1
                                        .WORD   BIT2
                                        .WORD   BIT3
                                        .WORD   BIT4
                                        .WORD   BIT5
                                        .WORD   BIT6
                                        .WORD   BIT8
                                        .WORD   BIT9
                                        .WORD   1576             ;GROWING 0
                                        .WORD   1574
                                        .WORD   1570
                                        .WORD   1560
                                        .WORD   1540
                                        .WORD   1500
                                        .WORD   1400
                                        .WORD   1576             ;SHIFT 0
                                        .WORD   1574
                                        .WORD   1566
                                        .WORD   1556
                                        .WORD   1536
                                        .WORD   1436
                                        .WORD   1136
                                        .WORD   76
                                        .WORD   6                ;GROWING 1
                                        .WORD   16
                                        .WORD   36
                                        .WORD   76
                                        .WORD   176              ;
                                        .WORD   576
                                        .WORD   1576
                                CSEND:  .WORD   0
                                ERPOINT: .WORD  0
                                ERCOUNT: .BLKW  64.
                                HDRBUF: .BLKW   160.
```

```
003650                              ENDMOD

003650                              BGNMOD  GLBTXT
                                    .SBTTL  GLOBAL TEXT

003650  116     117     040     NORES:  .ASCIZ  /NO CONTROLLER/
003666  116     117     040     NODRY:  .ASCIZ  /NO DRIVE CONNECTED/
003711  040     104     122     DEMES:  .ASCIZ  / DRV/
003716  040     116     130     NXMMES: .ASCIZ  / NXM/
003723  040     117     120     OPIMES: .ASCIZ  / OPI/
003730  040     110     103     HCRCMES:.ASCIZ  / HCRC/
003736  040     110     116     HNFMES: .ASCIZ  / HNF/
003743  040     104     103     DCKMES: .ASCIZ  / DCK/
003750  040     104     114     DLTMES: .ASCIZ  / DLT/
003755  015     012     000     MSCRLF: .ASCIZ  <15><12>
003760  015     000             LF:     .ASCIZ  <15>
003762  040     103     117     COMP:   .ASCIZ  / COMP/
003770  106     117     122     OPIERR: .ASCIZ  /FORCED OPI(GET STATUS) CAUSED OTHER ERRORS/
004043  116     117     117     NOPMES: .ASCIZ  /NOOP OPERATION-FLAG MODE/
004074  116     117     117     NOPINT: .ASCIZ  /NOOP OPERATION-INTR. MODE/
004126  127     122     111     WCKMES: .ASCIZ  /WRITE CHECK OPERATION-FLAG MODE/
004166  127     122     111     WCKINT: .ASCIZ  /WRITE CHECK OPERATION-INTR. MODE/
004227  122     105     101     RHDMES: .ASCIZ  /READ HEADER OPERATION-FLAG MODE/
004267  122     105     101     RHDINT: .ASCIZ  /READ HEADER OPERATION-INTR. MODE/
004330  123     105     105     SEKMES: .ASCIZ  /SEEK OPERATION-FLAG MODE/
004361  123     105     105     SEKINT: .ASCIZ  /SEEK OPERATION-INTR. MODE/
004413  107     105     124     GSTMES: .ASCIZ  /GET STATUS OPERATION-FLAG MODE/
004452  107     105     124     GSTINT: .ASCIZ  /GET STATUS OPERATION-INTR MODE/
004511  103     123     072     ARLCS:  .ASCIZ  /CS: /
004516  040     102     101     ARLBA:  .ASCIZ  / BA: /
004524  040     104     101     ARLDA:  .ASCIZ  / DA: /
004532  040     115     120     ARLMP:  .ASCIZ  / MP: /
004540  102     105     106     BEREG:  .ASCIZ  /BEFORE COMMAND: /
004561  124     111     115     AFREG:  .ASCIZ  /TIME OF ERROR:  /
004602  103     117     116     CRTIM:  .ASCIZ  /CONTROLLER TIMED OUT/
004627  104     122     111     DRTIM:  .ASCIZ  /DRIVE READY TIMED OUT/
004655  103     101     116     EM1:    .ASCIZ  /CAN NOT ADDRESS RLCS/
004702  103     101     116     EM2:    .ASCIZ  /CAN NOT ADDRESS RLBA/
004727  103     101     116     EM3:    .ASCIZ  /CAN NOT ADDRESS RLDA/
004754  103     101     116     EM4:    .ASCIZ  /CAN NOT ADDRESS RLMP/
005001  122     114     103     EM5:    .ASCIZ  %RLCS READ/WRITE ERROR (BIT 0 DON'T CARE)%
005052  122     114     102     EM6:    .ASCIZ  %RLBA READ/WRITE ERROR%
005100  122     114     104     EM7:    .ASCIZ  %RLDA READ/WRITE ERROR%
005126  117     120     111     EM11:   .ASCIZ  /OPI WOULD NOT GENERATE INTERRUPT/
005167  116     117     040     EM13:   .ASCIZ  /NO INTERRUPT FROM NOOP(0)/
005221  116     117     117     EM14:   .ASCIZ  /NOOP(0) MODIFIED RLMP/
005247  116     117     117     EM15:   .ASCIZ  /NOOP(0) MODIFIED RLBA/
005275  116     117     117     EM16:   .ASCIZ  /NOOP(0) MODIFIED RLDA/
005323  111     116     124     EM17:   .ASCIZ  /INTERRUPT PRIORITY FAILURE/
005356  107     105     124     EM30:   .ASCIZ  /GET STATUS WOULD NOT INTERRUPT/
005415  107     105     124     EM30A:  .ASCIZ  /GET STATUS SHOULD NOT INTERRUPT/
005455  122     114     115     EM32:   .ASCIZ  /RLMP CONTAINED WRONG STATUS/
005511  117     120     111     EM33:   .ASCIZ  /OPI DID NOT SET-GSTAT WITHOUT GS BIT/
005556  117     120     111     EM34:   .ASCIZ  /OPI DID NOT SET-GSTAT WITHOUT GS AND MK BITS/
005633  122     105     101     EM37:   .ASCIZ  /READ HEADER WOULD NOT INTERRUPT/
005673  102     101     104     EM41:   .ASCIZ  /BAD CYLINDER OR HEAD SELECT IN REPEATED READ HEADER TEST/
005764  102     101     104     EM42:   .ASCIZ  /BAD HEADER CRC ON READ HEADER/
```

```
354 006022  123  105  103  EM43:   .ASCIZ  /SECTOR ADDRESS OUT OF SEQUENCE DURING CONSECUTIVE READ HEADERS/
355 006121  127  122  111  EM44:   .ASCIZ  /WRITING RLMP MODIFIED RLCS/
356 006154  127  122  111  EM45:   .ASCIZ  /WRITING RLMP MODIFIED RLBA/
357 006207  127  122  111  EM46:   .ASCIZ  /WRITING RLMP MODIFIED RLDA/
358 006242  123  105  105  EM47:   .ASCIZ  /SEEK WOULD NOT INTERRUPT/
359 006273  104  122  111  EM52:   .ASCIZ  /DRIVE READY CAUSED EXTRANEOUS INTERRUPT/
360 006343  102  101  104  EM54:   .ASCIZ  /BAD SEEK-TEST OF DIFFENCE WORD/
361 006402  102  101  104  EM55:   .ASCIZ  /BAD HEAD SELECT VIA RD HDR/
362 006435  102  101  104  EM56:   .ASCIZ  /BAD HEAD SELECT VIA GET STATUS/
363 006474  114  117  101  EM57:   .ASCII  /LOADING RLDA BEFORE DRIVE READY ON SEEK/<15><12>
364 006545  104  122  111          .ASCIZ  /DRIVE READY DID NOT SET/
365 006575  102  111  124  EM61:   .ASCIZ  /BIT SET INSTRUCTION ON RLCS YIELDED WRONG RESULT/
366 006656  102  111  124  EM62:   .ASCIZ  /BIT CLEAR INSTRUCTION ON RLCS YIELDED WRONG RESULT/
367 006741  102  111  124  EM63:   .ASCIZ  /BIT SET INSTRUCTION ON RLBA YIELDED WRONG RESULT/
368 007022  102  111  124  EM64:   .ASCIZ  /BIT CLEAR INSTRUCTION ON RLBA YIELDED WRONG RESULT/
369 007105  102  111  124  EM65:   .ASCIZ  /BIT SET INSTRUCTION ON RLDA YIELDED WRONG RESULT/
370 007166  102  111  124  EM66:   .ASCIZ  /BIT CLEAR INSTRUCTION ON RLDA YIELDED WRONG RESULT/
371 007251  102  125  123  EM67:   .ASCIZ  /BUS RESET DID NOT CLEAR RLCS/
372 007306  102  125  123  EM70:   .ASCIZ  /BUS RESET DID NOT CLEAR RLBA/
373 007343  102  125  123  EM71:   .ASCIZ  /BUS RESET DID NOT CLEAR RLDA/
374 007400  127  122  111  EM72:   .ASCIZ  /WRITING RLCS MODIFIED RLBA/
375 007433  127  122  111  EM73:   .ASCIZ  /WRITING RLCS MODIFIED RLDA/
376 007466  127  122  111  EM74:   .ASCIZ  /WRITING RLBA MODIFED RLCS/
377 007520  127  122  111  EM75:   .ASCIZ  /WRITING RLBA MODIFED RLDA/
378 007552  127  122  111  EM76:   .ASCIZ  /WRITING RLDA MODIFIED RLCS/
379 007605  127  122  111  EM77:   .ASCIZ  /WRITING RLDA MODIFIED RLBA/
380 007640  122  114  103  EM101:  .ASCIZ  /RLCS CONTAINED FOLLOWING ERROR(S):  /
381 007705               EM102:  .BLKB   120.
382
383                      .EVEN
384
388 010076                       ENDMOD
389
390                      .SBTTL  GLOBAL ERRORS
391
392 010076               BGNMOD  GLBERR
393
394 010076               BGNMSG  ERR0
395
396 010076  004737  010422       JSR     PC,LINE1
397 010102  004737  010456       JSR     PC,LINE2
398
399 010106  004537  012516       JSR     R5,CKERLT       ;CHECK ERROR LIMIT
400 010112                       ENDMSG
    010112               L10000:
    010112  104023               EMT     C$MSG
401
402 010114               BGNMSG  ERR1
403
404 010114  004737  010422       JSR     PC,LINE1
405
406 010120  004537  012516       JSR     R5,CKERLT       ;CHECK ERROR LIMIT
407 010124                       ENDMSG
    010124               L10001:
    010124  104023               EMT     C$MSG
```

```
010126                          BGNMSG  ERR2
010126  004737  010422                  JSR     PC,LINE1
010132                                  PRINTB  #FRMT4,GDDAT,BDDAT
010132  013746  002242                  MOV     BDDAT,-(SP)
010136  013746  002240                  MOV     GDDAT,-(SP)
010142  012746  011100                  MOV     #FRMT4,-(SP)
010146  012746  000003                  MOV     #3,-(SP)
010152  010600                          MOV     SP,R0
010154  104014                          EMT     C$PNTB
010156  062706  000010                  ADD     #10,SP
010162  004537  012516                  JSR     R5,CKERLT       ;CHECK ERROR LIMIT
010166                                  ENDMSG
010166                          L10002:
010166  104023                          EMT     C$MSG
010170                          BGNMSG  ERR3
010170  004737  010422                  JSR     PC,LINE1
010174  004737  010456                  JSR     PC,LINE2
010200                                  PRINTB  #FRMT5,TMPO,BDDAT,GDDAT
010200  013746  002240                  MOV     GDDAT,-(SP)
010204  013746  002242                  MOV     BDDAT,-(SP)
010210  013746  002232                  MOV     TMPO,-(SP)
010214  012746  011136                  MOV     #FRMT5,-(SP)
010220  012746  000004                  MOV     #4,-(SP)
010224  010600                          MOV     SP,R0
010226  104014                          EMT     C$PNTB
010230  062706  000012                  ADD     #12,SP
010234  004537  012516                  JSR     R5,CKERLT       ;CHECK ERROR LIMIT
010240                                  ENDMSG
010240                          L10003:
010240  104023                          EMT     C$MSG
010242                          BGNMSG  ERR4
010242  004737  010422                  JSR     PC,LINE1
010246  004737  010456                  JSR     PC,LINE2
010252                                  PRINTB  #FRMT4,GDDAT,BDDAT
010252  013746  002242                  MOV     BDDAT,-(SP)
010256  013746  002240                  MOV     GDDAT,-(SP)
010262  012746  011100                  MOV     #FRMT4,-(SP)
010266  012746  000003                  MOV     #3,-(SP)
010272  010600                          MOV     SP,R0
010274  104014                          EMT     C$PNTB
010276  062706  000010                  ADD     #10,SP
010302  004537  012516                  JSR     R5,CKERLT       ;CHECK ERROR LIMIT
010306                                  ENDMSG
010306                          L10004:
010306  104023                          EMT     C$MSG
010310                          BGNMSG  ERR5
```

```
010310  004737  010422                  JSR     PC,LINE1
010314  004537  012516                  JSR     R5,CKERLT       ;CHECK ERROR LIMIT
010320                                  ENDMSG
010320                          L10005:
010320  104023                          EMT     C$MSG
010322                          BGNMSG  ERR6
010322  004737  010422                  JSR     PC,LINE1
010326  004737  010700                  JSR     PC,LINE3
010332  004737  010456                  JSR     PC,LINE2
010336                          1$:     PRINTB  #FRMT99
010336  012746  011133                  MOV     #FRMT99,-(SP)
010342  012746  000001                  MOV     #1,-(SP)
010346  010600                          MOV     SP,R0
010350  104014                          EMT     C$PNTB
010352  062706  000004                  ADD     #4,SP
010356  004537  012516                  JSR     R5,CKERLT       ;CHECK ERROR LIMIT
010362                                  ENDMSG
010362                          L10006:
010362  104023                          EMT     C$MSG
010364                          BGNMSG  ERR7
010364  004737  010422                  JSR     PC,LINE1
010370                                  PRINTB  #FRMT6,BDDAT
010370  013746  002242                  MOV     BDDAT,-(SP)
010374  012746  011207                  MOV     #FRMT6,-(SP)
010400  012746  000002                  MOV     #2,-(SP)
010404  010600                          MOV     SP,R0
010406  104014                          EMT     C$PNTB
010410  062706  000006                  ADD     #6,SP
010414  004537  012516                  JSR     R5,CKERLT
010420                                  ENDMSG
010420                          L10007:
010420  104023                          EMT     C$MSG
010422                          LINE1:  PRINTB  #FRMT1,RLCS,<B,DRIVE+1>
010422  005046                          CLR     -(SP)
010424  153716  002153                  BISB    DRIVE+1,(SP)
010430  013746  002134                  MOV     RLCS,-(SP)
010434  012746  010752                  MOV     #FRMT1,-(SP)
010440  012746  000003                  MOV     #3,-(SP)
010444  010600                          MOV     SP,R0
010446  104014                          EMT     C$PNTB
010450  062706  000010                  ADD     #10,SP
010454  000207                          RTS     PC
010456                          LINE2:  PRINTB  #FRMT2,#BEREG,#ARLCS,B.CS,#ARLBA,B.BA
010456  013746  002156                  MOV     B.BA,-(SP)
010462  012746  004516                  MOV     #ARLBA,-(SP)
010466  013746  002154                  MOV     B.CS,-(SP)
```

```
        010472  012746  004511                  MOV     #ARLCS,-(SP)
        010476  012746  004540                  MOV     #BEREG,-(SP)
        010502  012746  011012                  MOV     #FRMT2,-(SP)
        010506  012746  000006                  MOV     #6,-(SP)
        010512  010600                          MOV     SP,R0
        010514  104014                          EMT     C$PNTB
        010516  062706  000016                  ADD     #16,SP
    59  010522                                  PRINTB  #FRMT2A,#ARLDA,B.DA,#ARLMP,B.MP
        010522  013746  002162                  MOV     B.MP,-(SP)
        010526  012746  004532                  MOV     #ARLMP,-(SP)
        010532  013746  002160                  MOV     B.DA,-(SP)
        010536  012746  004524                  MOV     #ARLDA,-(SP)
        010542  012746  011031                  MOV     #FRMT2A,-(SP)
        010546  012746  000005                  MOV     #5,-(SP)
        010552  010600                          MOV     SP,R0
        010554  104014                          EMT     C$PNTB
        010556  062706  000014                  ADD     #14,SP
    60  010562                                  PRINTB  #FRMT2,#AFREG,#ARLCS,E.CS,#ARLBA,E.BA
        010562  013746  002170                  MOV     E.BA,-(SP)
        010566  012746  004516                  MOV     #ARLBA,-(SP)
        010572  013746  002166                  MOV     E.CS,-(SP)
        010576  012746  004511                  MOV     #ARLCS,-(SP)
        010602  012746  004561                  MOV     #AFREG,-(SP)
        010606  012746  011012                  MOV     #FRMT2,-(SP)
        010612  012746  000006                  MOV     #6,-(SP)
        010616  010600                          MOV     SP,R0
        010620  104014                          EMT     C$PNTB
        010622  062706  000016                  ADD     #16,SP
    61  010626                                  PRINTB  #FRMT2B,#ARLDA,E.DA,#ARLMP,E.MP,E.MP1,E.MP2
        010626  013746  002200                  MOV     E.MP2,-(SP)
        010632  013746  002176                  MOV     E.MP1,-(SP)
        010636  013746  002174                  MOV     E.MP,-(SP)
        010642  012746  004532                  MOV     #ARLMP,-(SP)
        010646  013746  002172                  MOV     E.DA,-(SP)
        010652  012746  004524                  MOV     #ARLDA,-(SP)
        010656  012746  011044                  MOV     #FRMT2B,-(SP)
        010662  012746  000007                  MOV     #7,-(SP)
        010666  010600                          MOV     SP,R0
        010670  104014                          EMT     C$PNTB
        010672  062706  000020                  ADD     #20,SP
    62  010676  000207                          RTS     PC
    63
    64  010700                          LINE3:  PRINTB  #FRMT3,#EM101
        010700  012746  007640                  MOV     #EM101,-(SP)
        010704  012746  011073                  MOV     #FRMT3,-(SP)
        010710  012746  000002                  MOV     #2,-(SP)
        010714  010600                          MOV     SP,R0
        010716  104014                          EMT     C$PNTB
        010720  062706  000006                  ADD     #6,SP
    65  010724                                  PRINTB  #FRMT3,#EM102
        010724  012746  007705                  MOV     #EM102,-(SP)
        010730  012746  011073                  MOV     #FRMT3,-(SP)
        010734  012746  000002                  MOV     #2,-(SP)
        010740  010600                          MOV     SP,R0
        010742  104014                          EMT     C$PNTB
        010744  062706  000006                  ADD     #6,SP
    66  010750  000207                          RTS     PC
```

```
     67
     71
     72 010752      045      101      103  FRMT1:   .ASCIZ  /%ACONTROLLER: %O6%A  DRIVE: %O1/
     73 011012      045      116      045  FRMT2:   .ASCIZ  /%N%T%T%O6%T%O6/
     74 011031      045      124      045  FRMT2A:  .ASCIZ  /%T%O6%T%O6/
     75 011044      045      124      045  FRMT2B:  .ASCIZ  /%T%O6%T%O6%A %O6%A %O6/
     76 011073      045      116      045  FRMT3:   .ASCIZ  /%N%T/
     77 011100      045      116      045  FRMT4:   .ASCII  /%N%AEXP'D: %O6%A REC'D: %O6/
     78 011133      045      116      000  FRMT99:  .ASCIZ  /%N/
     79 011136      045      116      045  FRMT5:   .ASCIZ  /%N%ALAST: %O6%A PRES: %O6%A EXP'D: %O6%N/
     80 011207      045      116      045  FRMT6:   .ASCIZ  /%N%AAT PROCESSOR LEVEL %O6%N/
     81 011244      045      101      105  FRMT11:  .ASCIZ  /%AERROR LIMIT EXCEEDED-DROPPED%N/
     82 011305      045      116      045  FRMT12:  .ASCIZ  /%N%ADRIVE DID NOT RECOVER FROM POWER FAILURE%N/
     83 011364      045      116      045  FRMT13:  .ASCIZ  /%N%T%A - WILL NOT TEST%N/
     84
     85                                    .EVEN
     86
     87
     91
     92
     93
     94 011416                            ENDMOD
     95
     96 011416                            BGNMOD  HPTCODE
     97
     98 011416                            BGNHW
        011416  000006                              .WORD   L10010-L$HW/2
     99 011420  174400                              .WORD   174400          ;CSR
    100 011422  000160                              .WORD   160             ;VECTOR
    101 011424  000240                              .WORD   240             ;PRIORITY
    102 011426  000001                              .WORD   1               ;RL01 = 1
    103 011430  000000                              .WORD   0               ;DRIVE (BITS 8,9,10)
    104 011432  000001                              .WORD   1               ;RL11 = 1, RLV11 = 0
    105
    106 011434                           ENDHW
        011434                           L10010:
    107
    108 011434                           ENDMOD
    109
    110 011434                           BGNMOD  SPTCODE
    111
    112 011434                           BGNSW
        011434  000003                              .WORD   L10011-L$SW/2
    113
    114 011436  000000                   DROP:   .WORD   0
    115 011440  000012                   MERLMT: .WORD   10.
    116 011442  000000                   T.SIZE: .WORD   0
    117
    118 011444                           ENDSW
        011444                           L10011:
    119
    120 011444                           ENDMOD
    121
    122 011444                           BGNMOD  DSPCODE
    123
    124 011444                           DISPATCH        44
        011444  000054                              .WORD   44
```

```
                011446  013770                          .WORD   T1
                011450  014064                          .WORD   T2
                011452  014160                          .WORD   T3
                011454  014254                          .WORD   T4
                011456  014350                          .WORD   T5
                011460  014470                          .WORD   T6
                011462  014572                          .WORD   T7
                011464  014660                          .WORD   T8
                011466  015004                          .WORD   T9
                011470  015130                          .WORD   T10
                011472  015234                          .WORD   T11
                011474  015334                          .WORD   T12
                011476  015424                          .WORD   T13
                011500  015524                          .WORD   T14
                011502  015634                          .WORD   T15
                011504  015706                          .WORD   T16
                011506  015744                          .WORD   T17
                011510  016070                          .WORD   T18
                011512  016230                          .WORD   T19
                011514  016370                          .WORD   T20
                011516  016574                          .WORD   T21
                011520  016624                          .WORD   T22
                011522  017030                          .WORD   T23
                011524  017114                          .WORD   T24
                011526  017260                          .WORD   T25
                011530  017310                          .WORD   T26
                011532  017462                          .WORD   T27
                011534  017550                          .WORD   T28
                011536  017676                          .WORD   T29
                011540  017720                          .WORD   T30
                011542  020000                          .WORD   T31
                011544  020144                          .WORD   T32
                011546  020302                          .WORD   T33
                011550  020620                          .WORD   T34
                011552  020656                          .WORD   T35
                011554  020722                          .WORD   T36
                011556  021046                          .WORD   T37
                011560  021464                          .WORD   T38
                011562  021616                          .WORD   T39
                011564  021760                          .WORD   T40
                011566  022120                          .WORD   T41
                011570  022272                          .WORD   T42
                011572  022720                          .WORD   T43
                011574  023440                          .WORD   T44
125
126             011576                                  ENDMOD
127
128                                             .SBTTL  INITIALIZATION CODE
129             011576                          BGNMOD  INITCODE
130
131             011576                                  BGNINIT
132
133             011576                                  BRESET
                011576  104033                          EMT     C$RESET
134             011600                                  READEF  #EF.PWR                 ;POWER UP?????
                011600  012700  000034                  MOV     #EF.PWR,R0
                011604  104050                          EMT     C$REFG
```

```
                                                        BNCOMPLETE      NOPWR           ;NO,BRANCH
011606  103004                                          BCC     NOPWR
011610  013737  002012  002126                        MOV     L$UNIT,PWRFLG           ;YES, SET POWER FLAG
011616  000475                                        BR      CONT                    ;GO TO CONTINUE POINT
011620                                  NOPWR:  READEF  #EF.RESTART     ;RESTART?
011620  012700  000037                                MOV     #EF.RESTART,R0
011624  104050                                        EMT     C$REFG
011626                                                  BCOMPLETE START1
011626  103404                                        BCS     START1
011630                                                  READEF  #EF.START               ;START???
011630  012700  000040                                MOV     #EF.START,R0
011634  104050                                        EMT     C$REFG
011636                                                  BNCOMPLETE      CONTINUE
011636  103010                                        BCC     CONTINUE
011640  012700  002750                        START1: MOV     #ERCOUNT,R0
011644  012701  000100                                MOV     #64.,R1
011650  005020                                  1$:     CLR     (R0)+
011652  005301                                        DEC     R1
011654  001375                                        BNE     1$
011656  000407                                        BR      START

011660                                  CONTINUE:       READEF  #EF.CONTINUE    ;CONTINUE????
011660  012700  000036                                MOV     #EF.CONTINUE,R0
011664  104050                                        EMT     C$REFG
011666                                                  BCOMPLETE       CONT
011666  103451                                        BCS     CONT

011670  005737  002130                        NXT:    TST     UUT                     ;DONE ALL UUT'S
011674  001011                                        BNE     XXX                     ;NO
011676  012737  177777  002132                START:  MOV     #-1,UNITST
011704  013737  002012  002130                        MOV     L$UNIT,UUT
011712  012737  002746  002746                        MOV     #ERCOUNT-2,ERPOINT

011720  005237  002132                        XXX:    INC     UNITST
011724  062737  000002  002746                        ADD     #2,ERPOINT
011732  005337  002130                                DEC     UUT
011736                                          REST:   GPHARD  UNITST,R0
011736  013700  002132                                MOV     UNITST,R0
011742  104042                                        EMT     C$GPHRD
011744                                                  BCOMPLETE       1$
011744  103406                                        BCS     1$
011746  005737  002126                                TST     PWRFLG                  ;POWER FLAG TO 0
011752  001746                                        BEQ     NXT                     ;YES, DONT DEC IT
011754  005337  002126                                DEC     PWRFLG
011760  000743                                        BR      NXT                     ;GET NEXT ONE
011762  012037  002144                          1$:   MOV     (R0)+,BCSR
011766  012037  002150                                MOV     (R0)+,BVEC
011772  012037  002146                                MOV     (R0)+,BPRIOR
011776  012037  002264                                MOV     (R0)+,T.DRIVE
012002  012037  002152                                MOV     (R0)+,DRIVE
012006  012037  002266                                MOV     (R0)+,T.CNTLR   ;GET CONTROLLER TYPE

012012  013700  002144                          CONT: MOV     BCSR,R0
012016  010037  002134                                MOV     R0,RLCS
012022  062700  000002                                ADD     #2,R0
012026  010037  002136                                MOV     R0,RLBA
012032  062700  000002                                ADD     #2,R0
```

```
012036  010037  002140                   MOV     R0,RLDA
012042  062700  000002                   ADD     #2,R0
012046  010037  002142                   MOV     R0,RLMP
012052  005737  002126                   TST     PWRFLG
012056  001064                           BNE     5$
012060  005737  011442                   TST     T.SIZE                  ;DO WE WANT TO CHECK UNITS??
012064  001461                           BEQ     5$                      ;NO

012066  005037  002204                   CLR     TRPFLG                  ;CLR OUT TRAP FLAG
012072                                   SETVEC  ERRVEC,#TRPHAN,#340     ;SETUP VECTOR TO CATCH NON-EXIST
012072  012746  000340                   MOV     #340,-(SP)
012076  012746  013632                   MOV     #TRPHAN,-(SP)
012102  013746  002216                   MOV     ERRVEC,-(SP)
012106  012746  000003                   MOV     #3,-(SP)
012112  104037                           EMT     C$SVEC
012114  062706  000010                   ADD     #10,SP
012120  005777  170010                   TST     @RLCS                   ;ACCESS CONTROLLER
012124                                   CLRVEC  ERRVEC                  ;RELEASE VECTOR
012124  013700  002216                   MOV     ERRVEC,R0
012130  104036                           EMT     C$CVEC
012132  005737  002204                   TST     TRPFLG                  ;DID IT TRAP
012136  001404                           BEQ     7$                      ;NO, CHECK IT'S DRIVE
012140  012737  003650  002262           MOV     #NORES,WHY              ;SETUP ERR MESS
012146  000415                           BR      8$

012150  012777  000200  167756   7$:     MOV     #200,@RLCS              ;CONTROLLER READY
012156  053777  002152  167750           BIS     DRIVE,@RLCS             ;SELECT DRIVE
012164  032777  000001  167742           BIT     #1,@RLCS                ;DRIVE THERE
012172  001016                           BNE     5$                      ;YES
012174  012737  003666  002262           MOV     #NODRY,WHY              ;SETUP ERR MESS
012202                           8$:     PRINTB  #FRMT13,WHY
012202  013746  002262                   MOV     WHY,-(SP)
012206  012746  011364                   MOV     #FRMT13,-(SP)
012212  012746  000002                   MOV     #2,-(SP)
012216  010600                           MOV     SP,R0
012220  104014                           EMT     C$PNTB
012222  062706  000006                   ADD     #6,SP
012226  000434                           BR      6$

012230  005737  002126           5$:     TST     PWRFLG                  ;RECENT POWER FAILURE????
012234  001457                           BEQ     END                     ;NO

                                 ;THERE WAS A RECENT POWER FAILURE, THEREFORE WE WILL WAIT
                                 ;SIXTY SECONDS FOR THE DRIVE TO COME READY

012236  012701  000074                   MOV     #60.,R1                 ;SIXTY SECOND TIMEOUT
012242  012777  000200  167664           MOV     #200,@RLCS              ;SET CRDY
012250  053777  002152  167656           BIS     DRIVE,@RLCS             ;SET IN DRIVE SELECT
012256  032777  000001  167650   2$:     BIT     #DRDY,@RLCS             ;DRIVE READY???
012264  001023                           BNE     3$                      ;YES, THEN START TEST

012266                                   WAITMS  #10.                    ;WAIT A SECOND
012266  012700  000012                   MOV     #10.,R0
012272  104026                           EMT     C$WTM

012274  005301                           DEC     R1                      ;SIXTY SECONDS GONE BY
012276  001367                           BNE     2$                      ;NO, GO BACK
```

```
                                                PRINTB  #FRMT12                 ;DROPPING DRIVE
012300  012746  011305                          MOV     #FRMT12,-(SP)
012304  012746  000001                          MOV     #1,-(SP)
012310  010600                                  MOV     SP,R0
012312  104014                                  EMT     C$PNTB
012314  062706  000004                          ADD     #4,SP
012320  004737  010422          6$:             JSR     PC,LINE1                ;GIVE DRIVE INFO
012324                                          DODU    UNITST                  ;TELL SUPERVISOR TO DROP IT
012324  013700  002132                          MOV     UNITST,R0
012330  104053                                  EMT     C$DODU
012332                                          DOCLN                           ;FORCE AN ABORT
012332  104044                                  EMT     C$DCLN

012334  012777  000013  167576  3$:             MOV     #13,@RLDA               ;SETUP DR RST
012342  012777  000204  167564                  MOV     #204,@RLCS              ;GS FUNC
012350  053777  002152  167556                  BIS     DRIVE,@RLCS             ;SELECT DRIVE
012356  042777  000200  167550                  BIC     #200,@RLCS              ;ISSUE IT
012364  032777  000200  167542  4$:             BIT     #200,@RLCS              ;WAIT FOR READY
012372  001774                                  BEQ     4$

012374                          END:            SETVEC  BVEC,#INTSRV,#340
012374  012746  000340                          MOV     #340,-(SP)
012400  012746  013640                          MOV     #INTSRV,-(SP)
012404  013746  002150                          MOV     BVEC,-(SP)
012410  012746  000003                          MOV     #3,-(SP)
012414  104037                                  EMT     C$SVEC
012416  062706  000010                          ADD     #10,SP
012422  005037  002202                          CLR     PFLG                    ;CLR PROCESSOR FLAG
012426                                          READBUS                         ;Q-BUS
012426  104007                                  EMT     C$RDBU
012430                                          BNCOMPLETE      1$
012430  103002                                  BCC     1$
012432  005237  002202                          INC     PFLG                    ;NO, Q-BUS THEN
012436                          1$:
012436                                          ENDINIT
012436                          L10012:
012436  104011                                  EMT     C$INIT

012440                                          ENDMOD

012440                          BGNMOD  CLNCODE

012440                                          BGNCLN

012440                                          SETPRI  #PR107
012440  012700  000340                          MOV     #PR107,R0
012444  104041                                  EMT     C$SPRI

012446  032777  000200  167460  1$:             BIT     #CRDY,@RLCS
012454  001774                                  BEQ     1$

012456  042777  000100  167450                  BIC     #INTEN,@RLCS

012464                                          CLRVEC  BVEC
012464  013700  002150                          MOV     BVEC,R0
```

```
        012470  104036                          EMT     C$CVEC
256
257
258
259 012472  005737  002126                      TST     PWRFLG          ;TREAT POWER FAILURE
260 012476  001402                              BEQ     2$
261
262 012500  005337  002126                      DEC     PWRFLG
263
264 012504                              2$:
265 012504                                      ENDCLN
        012504                          L10013:
        012504  104012                          EMT     C$CLEAN
266
267 012506                                      ENDMOD
268
269
270
271 012506                              BGNMOD  DRPCODE
272
273 012506                                      BGNDU
274
275 012506  000240                              NOP
276
277 012510                                      ENDDU
        012510                          L10014:
        012510  104055                          EMT     C$DU
278
279 012512                              ENDMOD
280
281 012512                              BGNMOD  ADDCODE
282
283 012512                                      BGNAU
284
285 012512  000240                              NOP
286
287 012514                                      ENDAU
        012514                          L10015:
        012514  104054                          EMT     C$AU
288
289 012516                              ENDMOD
290
291
292                                     .SBTTL  GLOBAL SUBROUTINES
293
294 012516                              BGNMOD  GLBSUB
295
296 012516                              CKERLT: INLOOP
        012516  104020                          EMT     C$INLP
297 012520                                      BCOMPLETE       99$
        012520  103427                          BCS     99$
298 012522  005737  011436                      TST     DROP
299 012526  001424                              BEQ     99$
300 012530  005277  170212                      INC     @ERPOINT
301 012534  027737  170206  011440              CMP     @ERPOINT,MERLMT
302 012542  002416                              BLT     99$
303
```

```
                                        PRINTF  #FRMT11
012544  012746  011244                  MOV     #FRMT11,-(SP)
012550  012746  000001                  MOV     #1,-(SP)
012554  010600                          MOV     SP,R0
012556  104017                          EMT     C$PNTF
012560  062706  000004                  ADD     #4,SP
012564  004737  010422                  JSR     PC,LINE1
012570                                  DODU    UNITST          ;DROP THE UNIT
012570  013700  002132                  MOV     UNITST,R0
012574  104053                          EMT     C$DODU
012576                                  DOCLN
012576  104044                          EMT     C$DCLN
012600                          99$:
012600  000205                          RTS     R5


                        .SBTTL  ROUTINE TO CHECK FOR CONTROLLER ERRORS

                        ;*********************************************************************
                        ;*THIS ROUTINE WILL CHECK RLCS FOR ERRORS AND PRINT THEM
                        ;*ACCORDINGLY.  IT WILL MERGE THE ERROR PRINTOUT WITH THE TEST
                        ;*ERROR MESSAGE.
                        ;*
                        ;*EXAMPLE:   RLCS CONTAINED FOLLOWING ERROR(S):
                        ;*                  DRV   OPI   HCRC   HNF
                        ;*           SEEK UNDER INTERRUPT
                        ;*
                        ;*
                        ;*
                        ;*ROUTINE USES R0,R1 AND PICKS HEADER FROM R3
                        ;*
                        ;*      CALL    JSR     R5,CHERR
                        ;*
                        ;*
                        ;*

002164          CHERR:  CLR     DERFLG          ;CLEAR OUT DRIVE ERROR FLAG
176000  002166          BIT     #176000,E.CS    ;ANY ERRORS SET
001001                  BNE     199$            ;IF YES, INVESTIGATE
000205                  RTS     R5              ;NO, EXIT
002270  000004  199$:   CMP     TMPFNC,#GSTAT   ;FUNCTION-NOP, RESET, GETSTATUS
002401                  BLT     98$             ;YES, GO CHECK IF ONLY DRIVE ERROR
000414                  BR      1$              ;YES SERVICE ERROR
002270  000002  98$:    CMP     TMPFNC,#WRCHK
001410                  BEQ     1$
013700  002166          MOV     E.CS,R0         ;GET E.CS
042700  001777          BIC     #1777,R0        ;
022700  140000          CMP     #140000,R0      ;DRIVE ERROR ALONE?
001001                  BNE     1$              ;NO, GO SERVICE
000205          2$:     RTS     R5              ;YES, EXIT

012701  007705  1$:     MOV     #EM102,R1       ;GET START OF STRING
005737  002166          TST     E.CS            ;IS COMPOSITE ERROR SET?(BETTER BE)
100003                  BPL     99$             ;IT'S NOT SOMETHING IS WRONG
004537  013346          JSR     R5,FIX          ;YES, PUT "COMP" IN STRING
003762                  COMP                    ;"COMP"
```

```
012702  032737  040000  002166  99$:    BIT     #DERR,E.CS      ;DRIVE ERROR SET?
012710  001405                          BEQ     3$              ;NO, CONTINUE
012712  005237  002164                  INC     DERFLG          ;SET DRV ERROR FLAG
012716  004537  013346                  JSR     R5,FIX          ;YES, PUT ''DRV'' INTO STRING
012722  003711                          DEMES                   ;''DRV''
012724  032737  020000  002166  3$:     BIT     #NXM,E.CS       ;NON-EXISTENT MEMORY ERROR?
012732  001403                          BEQ     4$              ;NO, CONTINUE
012734  004537  013346                  JSR     R5,FIX          ;YES, PUT ''NXM'' INTO STRING
012740  003716                          NXMMES                  ;''NXM''
012742  032737  002000  002166  4$:     BIT     #OPI,E.CS       ;IS OPI SET?
012750  001422                          BEQ     6$              ;NO, GO CHECK BITS 11 & 12
012752  004537  013346                  JSR     R5,FIX          ;PUT ''OPI'' INTO STRING
012756  003723                          OPIMES                  ;''OPI''
012760  032737  004000  002166          BIT     #BIT11,E.CS     ;HEADERCRC ERROR?
012766  001403                          BEQ     5$              ;NO, GO CHECK HEADER NOT FOUND
012770  004537  013346                  JSR     R5,FIX          ;GO PUT ''HCRC'' IN STRING
012774  003730                          HCRCMES                 ;''HCRC''
012776  032737  010000  002166  5$:     BIT     #BIT12,E.CS     ;HEADER NOT FOUND?
013004  001422                          BEQ     8$              ;NO, GO PUT ''CRLF'' IN STRING
013006  004537  013346                  JSR     R5,FIX          ;PUT ''HNF'' IN STRING
013012  003736                          HNFMES                  ;''HNF''
013014  000416                          BR      8$              ;PUT ''CRLF'' IN STRING
013016  032737  004000  002166  6$:     BIT     #BIT11,E.CS     ;DATA CRC ERROR?
013024  001403                          BEQ     7$              ;NO, GO CHECK DATA LATE
013026  004537  013346                  JSR     R5,FIX          ;PUT ''DCK'' IN STRING
013032  003743                          DCKMES                  ;''DCK''
013034  032737  010000  002166  7$:     BIT     #BIT12,E.CS     ;DATA LATE ERROR?
013042  001403                          BEQ     8$              ;NO, GO PUT IN ''CRLF''
013044  004537  013346                  JSR     R5,FIX          ;PUT ''DLT'' IN STRING
013050  003750                          DLTMES                  ;''DLT''
013052  004537  013346          8$:     JSR     R5,FIX
013056  003755                          MSCRLF
013060  004537  013346                  JSR     R5,FIX
013064  000000                  RESTMS: .WORD   0               ;HEADER FROM TEST
013066  105011                          CLRB    (R1)            ;PUT TERMINATOR IN

013070                                  ERRDF   300.,LF,ERR6
013070  104462                          TRAP    T$ERCODE
013072  000454                          .WORD   300
013074  003760                          .WORD   LF
013076  010322                          .WORD   ERR6

013100  000205                          RTS     R5              ;EXIT ROUTINE

                                .SBTTL LOAD RLCS
                                ;*************************************************************
                                ;* ROUTINE TO LOAD RLCS WITH FUNCTION TO BE PERFORMED
                                ;*      CALL:   JSR     R5,LDFUNC
                                ;*              .WORD           ;BITS TO BE LOADED, FUNCTION
                                ;*                              ;AND INTR ENABLE ONLY
                                ;*
                                ;

013102  012537  002210          LDFUNC: MOV     (R5)+,LDCSR     ;GET BITS TO LOAD
013106  005737  002164                  TST     DERFLG
013112  001424                          BEQ     98$
013114  013746  002154                  MOV     B.CS,-(SP)
```

```
013120  012777  000013  167012          MOV     #13,@RLDA
013126  012737  000004  002154          MOV     #GSTAT,B.CS
013134  053737  002152  002154          BIS     DRIVE,B.CS
013142  013777  002154  166764          MOV     B.CS,@RLCS
013150  012637  002154                  MOV     (SP)+,B.CS
013154  032777  000200  166752  99$:    BIT     #200,@RLCS
013162  001774                          BEQ     99$
013164  010346                  98$:    MOV     R3,-(SP)          ;SAVE R3
013166  042737  177661  002210          BIC     #177661,LDCSR     ;CLEAR ALL BUT FUNC & INTR EN
013174  013737  002210  013320          MOV     LDCSR,FNDFNC      ;SAVE FUNCTION
013202  042737  000100  013320          BIC     #INTEN,FNDFNC     ;ONLY FUNCTION
013210  013737  013320  002270          MOV     FNDFNC,TMPFNC
013216  012703  013322                  MOV     #HDRLST,R3        ;GET HEADER LIST
013222  006237  013320                  ASR     FNDFNC            ;ALIGN TO RIGHT
013226  001404                          BEQ     2$
013230  022323                  1$:     CMP     (R3)+,(R3)+       ;BUMP R3 BY 4
013232  005337  013320                  DEC     FNDFNC            ;FOUND IT
013236  001374                          BNE     1$                ;NO,KEEP LOOKING
013240  032737  000100  002210  2$:     BIT     #INTEN,LDCSR      ;YES,DO WE WANT FLAG OR INTR
013246  001401                          BEQ     3$                ;FLAG BRANCH
013250  005723                          TST     (R3)+             ;INTR POINT TO THAT ONE
013252  011303                  3$:     MOV     (R3),R3           ;SET HEADER
013254  010337  013064                  MOV     R3,RESTMS         ;SET UP HEADER
013260  053737  002152  002210          BIS     DRIVE,LDCSR       ;SELECT DRIVE
013266  052737  000200  002210  4$:     BIS     #200,LDCSR        ;CONTROLLER READY
013274  013777  002210  166632          MOV     LDCSR,@RLCS
013302  004537  013360                  JSR     R5,BEFORE
013306  042777  000200  166620  5$:     BIC     #200,@RLCS
013314  012603                          MOV     (SP)+,R3          ;RESTORE R3
013316  000205                          RTS     R5                ;EXIT

013320  000000                  FNDFNC: .WORD   0

013322  004043                  HDRLST: NOPMES
013324  004074                          NOPINT
013326  004126                          WCKMES
013330  004166                          WCKINT
013332  004413                  OKHDR:  GSTMES
013334  004452                          GSTINT
013336  004330                          SEKMES
013340  004361                          SEKINT
013342  004227                          RHDMES
013344  004267                          RHDINT

                                ;*************************************************************
                                ;*ROUTINE TO MOVE ASCII STRINGS
                                ;*USES REGISTERS R1 - WHERE STRING IS BEING BUILT
                                ;*
                                ;*      CALL    JSR     R5,FIX
                                ;*              .WORD             ;ADDRESS OF STRING TO MOVE

013346  012500                  FIX:    MOV     (R5)+,R0          ;GET ADDRESS AND MOVE RETURN
013350  112021                  1$:     MOVB    (R0)+,(R1)+       ;GET BYTE AND UPDATE
013352  001376                          BNE     1$                ;WATCH 0 BYTE TERMINATOR
013354  105741                          TSTB    -(R1)             ;BACK UP OVER ZERO BYTE
013356  000205                          RTS     R5                ;EXIT
```

```
                                        ;LOAD REGISTERS BEFORE FUNCTION
                                        ;CALL:  JSR     R5,BEFORE

013360  017737  166550  002154  BEFORE: MOV     @RLCS,B.CS      ;READ CS
013366  017737  166544  002156          MOV     @RLBA,B.BA      ;READ BA
013374  017737  166540  002160          MOV     @RLDA,B.DA      ;READ DA
013402  017737  166534  002162          MOV     @RLMP,B.MP      ;READ MP
013410  000205                          RTS     R5

                                        ;LOAD REGISTERS AT ERROR
                                        ;CALL:  JSR     R5,AFTER

013412  017737  166516  002166  AFTER:  MOV     @RLCS,E.CS      ;READ CS
013420  017737  166512  002170          MOV     @RLBA,E.BA      ;READ BA
013426  017737  166506  002172          MOV     @RLDA,E.DA      ;READ DA
013434  017737  166502  002174          MOV     @RLMP,E.MP      ;READ MP
013442  017737  166474  002176          MOV     @RLMP,E.MP1     ;READ MP
013450  017737  166466  002200          MOV     @RLMP,E.MP2     ;READ MP
013456  000205                          RTS     R5


                                        .SBTTL  ROUTINE TO CALCULATE CRC

                                        ;ROUTINE WILL CALCULATE A CRC-16 CRC ON A WORD OF
                                        ;1-16 BITS IN LENGTH, RESULT IS RETURNED IN "CALBCC"
                                        ;
                                        ;       CALL:   JSR     R5,SIMBCC
                                        ;               .WORD           ;NUMBER OF BITS (1-16)
                                        ;               .WORD           ;DATA FOR CRC CALCULATION
                                        ;               .WORD           ;PREVIOUS OR STARTING CRC
                                        ;                               (SHOULD BE ZEROED FOR START)
                                        ;       ROUTINE USES R0,R1,R2

013460  010046          SIMBCC: MOV     R0,-(SP)        ;SAVE R0
013462  010146                  MOV     R1,-(SP)        ;SAVE R1
013464  010246                  MOV     R2,-(SP)        ;SAVE R2
013466  012537  002224          MOV     (R5)+,TEMP2     ;GET NUMBER OF BITS
013472  012537  002226          MOV     (R5)+,TEMP3     ;GET DATA FOR CRC CALCULATION
013476  012537  002230          MOV     (R5)+,TEMP4     ;GET STARTING CRC
013502  005037  002220  1$:     CLR     BCCFBK          ;
013506  013700  002230          MOV     TEMP4,R0        ;GET PRESENT CRC
013512  006037  002226          ROR     TEMP3           ;ROTATE NEW DATA
013516  005500                  ADC     R0              ;MERGE NEW WITH OLD
013520  032700  000001          BIT     #1,R0           ;BIT 0 SET
013524  001402                  BEQ     2$              ;IF NOT CONTINUE
013526  005137  002220          COM     BCCFBK          ;
013532  013700  002214  2$:     MOV     XPOLY,R0        ;GET CRC POLYNOMIAL (CRC-16)
013536  005100                  COM     R0              ;COMPLIMENT POLYNOMIAL
013540  040037  002220          BIC     R0,BCCFBK
013544  000241                  CLC                     ;CLEAR CARRY
013546  006037  002230          ROR     TEMP4
013552  013700  002220          MOV     BCCFBK,R0
013556  013701  002230          MOV     TEMP4,R1
013562  010102                  MOV     R1,R2
```

```
                                        BIC     R1,R0
                                        BIC     BCCFBK,R2
                                        BIS     R2,R0
                                        BIC     XPOLY,TEMP4
                                        BIS     R0,TEMP4
                                        DEC     TEMP2
                                        BNE     1$
                                        MOV     TEMP4,CALBCC
                                        MOV     (SP)+,R2
                                        MOV     (SP)+,R1
                                        MOV     (SP)+,R0
                                        RTS     R5              ;RETURN


                        ;ROUTINE TO SET FLAG IF TRAP OCCURRED
                        ;"TRPHAN" IS IN LOCATION 4.


                        TRPHAN: INC     TRPFLG          ;INDICATE TRAP
                                RTI                     ;RETURN

                        BGNSRV

                        INTSRV: INC     INTFLG          ;INDICATE INTERRUPT

                        ENDSRV
                        L10016:
                                RTI

                        ;ROUTINE TO WAIT FOR DRIVE READY
                        WTDRDY: MOV     R1,-(SP)        ;SAVE R1
                                MOV     #2000.,R1       ;TIME OUT OF 200 MILLISECONDS
                        1$:     BIT     #DRDY,@RLCS     ;DRIVE READY?
                                BNE     2$              ;YES, EXIT

                                WAITUS  #1              ;WAIT A WHILE
                                MOV     #1,R0
                                EMT     C$WTU
                                DEC     R1              ;CHECK IF TIME UP
                                BNE     1$              ;NO, GO CHECK DRIVE READY

                                ERRDF   200.,DRTIM,ERR5 ;DRIVE READY DID NOT SET
                                TRAP    T$ERCODE
                                .WORD   200
                                .WORD   DRTIM
                                .WORD   ERR5

                        2$:     MOV     (SP)+,R1        ;RESTORE
                                RTS     R5              ;EXIT

                        ;ROUTINE TO WAIT FOR CONTROLLER READY
                        WTCRDY: MOV     R1,-(SP)        ;SAVE R1
                                MOV     #8000.,R1       ;WAIT 800 MILLISECONDS
                        1$:     BIT     #CRDY,@RLCS     ;CONTROLLER READY
                                BNE     2$              ;YES, EXIT
                                WAITUS  #1              ;WAIT A WHILE
```

```
        013730  012700  000001                  MOV     #1,R0
        013734  104027                          EMT     C$WTU
    569 013736  005301                          DEC     R1              ;CHECK IF TIME UP
    570 013740  001367                          BNE     1$              ;NO GO BACK
    571
    572 013742  004537  013412                  JSR     R5,AFTER        ;GET REGISTERS
    573
    574 013746                                  ERRDF   100.,CRTIM,ERR6 ;CONTROLLER TIMED OUT
        013746  104462                          TRAP    T$ERCODE
        013750  000144                          .WORD   100
        013752  004602                          .WORD   CRTIM
        013754  010322                          .WORD   ERR6
    575
    576 013756  000402                          BR      3$              ;EXIT
    577
    578 013760  004537  013412          2$:     JSR     R5,AFTER        ;GET REGISTERS
    579 013764  012601                  3$:     MOV     (SP)+,R1
    580 013766  000205                          RTS     R5              ;EXIT
    581
    582
    583
    584 013770                                  ENDMOD
    585
    586
    587
    588                             .SBTTL  **TEST 1** - RLCS ADDRESSABILITY
    589
    590 013770                      BGNTST                              ;****START OF TEST****
    591 013770                      STARS
                                    ;;*****************************************************************
    592                             ;TEST TO SEE IF WE CAN ADDRESS THE CONTROL
    593                             ;AND STATUS REGISTER.  IF WE TRAP WE WILL REPORT
    594                             ;THE ERROR AND ABORT.  AFTER THIS TEST WE ONLY KNOW
    595                             ;THAT WE CAN ADDRESS THE REGISTER.
    596 013770                      STARS
                                    ;;*****************************************************************
    597
    598
    599 013770  005037  002204          1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
    600 013774                      2$:     SETVEC  ERRVEC,#TRPHAN,#340  ;SET TO CATCH TRAP
        013774  012746  000340                  MOV     #340,-(SP)
        014000  012746  013632                  MOV     #TRPHAN,-(SP)
        014004  013746  002216                  MOV     ERRVEC,-(SP)
        014010  012746  000003                  MOV     #3,-(SP)
        014014  104037                          EMT     C$SVEC
        014016  062706  000010                  ADD     #10,SP
    601
    602 014022  005777  166106                  TST     @RLCS           ;ADDRESS RLCS
    603 014026                                  CLRVEC  ERRVEC          ;RELEASE TRAP VECTOR
        014026  013700  002216                  MOV     ERRVEC,R0
        014032  104036                          EMT     C$CVEC
    604 014034  005737  002204                  TST     TRPFLG          ;TRAP OCCURRED???
    605 014040  001407                          BEQ     3$              ;NO, IKAY PROCEED
    606 014042  013737  002134  002240          MOV     RLCS,GDDAT      ;SET UP ERROR DATA
    607
    608 014050                                  ERRSF   0.,EM1,ERR1     ;BUS TIMEOUT IN ADDRESSING RLCS
        014050  104461                          TRAP    T$ERCODE
```

```
014052  000000                          .WORD   0
014054  004655                          .WORD   EM1
014056  010114                          .WORD   ERR1
609 014060                      3$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
014060  104006                          EMT     C$CLP1
610 014062                      ENDTST                          ;****END OF TEST****
014062                          L10017:
014062  104001                          EMT     C$ETST
611
612
613                             .SBTTL  **TEST 2** - RLBA ADDRESSABILITY
614
615 014064                      BGNTST                          ;****START OF TEST****
616
617
618 014064                      STARS
                                ;;**************************************************************
619                             ;TEST TO SEE IF WE CAN ADDRESS THE BUS ADDRESS
620                             ;REGISTER.  IF WE TRAP WE WILL REPORT THE ERROR
621                             ;AND ABORT.  AFTER THIS TEST WE ONLY KNOW THAT
622                             ;WE CAN ADDRESS THE REGISTER.
623 014064                      STARS
                                ;;**************************************************************
624
625 014064  005037  002204      1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
626 014070                      2$:     SETVEC  ERRVEC,#TRPHAN,#340  ;SET TO CATCH TRAP
014070  012746  000340                  MOV     #340,-(SP)
014074  012746  013632                  MOV     #TRPHAN,-(SP)
014100  013746  002216                  MOV     ERRVEC,-(SP)
014104  012746  000003                  MOV     #3,-(SP)
014110  104037                          EMT     C$SVEC
014112  062706  000010                  ADD     #10,SP
627
628 014116  005777  166014              TST     @RLBA           ;ADDRESS RLBA
629 014122                              CLRVEC  ERRVEC          ;RELEASE TRAP VECTOR
014122  013700  002216                  MOV     ERRVEC,R0
014126  104036                          EMT     C$CVEC
630 014130  005737  002204              TST     TRPFLG          ;TRAP OCCURRED???
631 014134  001407                      BEQ     3$              ;NO, CONTINUE
632 014136  013737  002136  002240      MOV     RLBA,GDDAT      ;SETUP ERROR DATA
633
634 014144                              ERRSF   1.,EM2,ERR1     ;BUS TIMEOUT IN ADDRESSING RLBA
014144  104461                          TRAP    T$ERCODE
014146  000001                          .WORD   1
014150  004702                          .WORD   EM2
014152  010114                          .WORD   ERR1
635 014154                      3$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
014154  104006                          EMT     C$CLP1
636 014156                      ENDTST                          ;****END OF TEST****
014156                          L10020:
014156  104001                          EMT     C$ETST
637
638
639                             .SBTTL  **TEST 3** - RLDA ADDRESSABILITY
640
641 014160                      BGNTST                          ;****START OF TEST****
642 014160                      STARS
```

```
                                ;;*****************************************************************
                                ;TEST TO SEE IF WE CAN ADDRESS THE DISK ADDRESS
                                ;REGISTER IF WE TRAP WE WILL REPORT THE ERROR
                                ;AND ABORT.  AFTER THIS TEST WE ONLY KNOW THAT
                                ;WE CAN ADDRESS THE REGISTER.
014160                          STARS
                                ;;*****************************************************************

014160  005037  002204        1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
014164                        2$:     SETVEC  ERRVEC,#TRPHAN,#340  ;SET TO CATCH TRAP
014164  012746  000340                MOV     #340,-(SP)
014170  012746  013632                MOV     #TRPHAN,-(SP)
014174  013746  002216                MOV     ERRVEC,-(SP)
014200  012746  000003                MOV     #3,-(SP)
014204  104037                       EMT     C$SVEC
014206  062706  000010                ADD     #10,SP

014212  005777  165722                TST     @RLDA           ;ADDRESS RLDA
014216                                CLRVEC  ERRVEC          ;RELEASE TRAP VECTOR
014216  013700  002216                MOV     ERRVEC,R0
014222  104036                       EMT     C$CVEC
014224  005737  002204                TST     TRPFLG          ;TRAP OCCURRED???
014230  001407                        BEQ     3$              ;NO, CONTINUE

014232  013737  002140  002240        MOV     RLDA,GDDAT      ;SETUP ERROR INFO
014240                                ERRSF   2.,EM3,ERR1     ;BUS TIMEOUT IN ADDRESSING RLDA
014240  104461                       TRAP    T$ERCODE
014242  000002                       .WORD   2
014244  004727                       .WORD   EM3
014246  010114                       .WORD   ERR1
014250                        3$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
014250  104006                       EMT     C$CLP1
014252                        ENDTST                          ;****END OF TEST****
014252                        L10021:
014252  104001                       EMT     C$ETST


                                .SBTTL  **TEST 4** - RLMP ADDRESSABILITY

014254                          BGNTST                          ;****START OF TEST****
014254                          STARS
                                ;;*****************************************************************
                                ;TEST TO SEE IF WE CAN ADDRESS THE MULTIPURPOSE
                                ;REGISTER.  IF WE TRAP WE WILL REPORT THE ERROR AND
                                ;ABORT.  AFTER THIS TEST WE ONLY KNOW THAT WE CAN
                                ;ADDRESS THE REGISTER.
014254                          STARS
                                ;;*****************************************************************


014254  005037  002204        1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
014260                        2$:     SETVEC  ERRVEC,#TRPHAN,#340  ;SET UP TO CATCH TRAP
014260  012746  000340                MOV     #340,-(SP)
014264  012746  013632                MOV     #TRPHAN,-(SP)
014270  013746  002216                MOV     ERRVEC,-(SP)
014274  012746  000003                MOV     #3,-(SP)
```

```
014300  104037                          EMT     C$SVEC
014302  062706  000010                  ADD     #10,SP
677
678 014306  005777  165630              TST     @RLMP           ;ADDRESS RLMP
679 014312                              CLRVEC  ERRVEC          ;RELEASE TRAP VECTOR
    014312  013700  002216              MOV     ERRVEC,R0
    014316  104036                      EMT     C$CVEC
680 014320  005737  002204              TST     TRPFLG          ;TRAP OCCURRED???
681 014324  001407                      BEQ     3$              ;NO, CONTINUE
682 014326  013737  002142  002240      MOV     RLMP,GDDAT      ;SET UP ERROR INFO
683
684 014334                              ERRSF   3.,EM4,ERR1     ;BUS TIMEOUT IN ADDRESSING RLMP
    014334  104461                      TRAP    T$ERCODE
    014336  000003                      .WORD   3
    014340  004754                      .WORD   EM4
    014342  010114                      .WORD   ERR1
685 014344                      3$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
    014344  104006                      EMT     C$CLP1
686 014346                      ENDTST                          ;****END OF TEST****
    014346                      L10022:
    014346  104001                      EMT     C$ETST
687
688
689                             .SBTTL  **TEST 5** - READ WRITE OF RLCS
690
691 014350                      BGNTST                          ;****START OF TEST****
692
693
694
695 014350                      STARS
                                ;;**********************************************************************
696                             ;TEST THAT WE CAN WRITE/READ BITS 8,9 AND BITS 6-1
697                             ;OF THE CONTROL AND STATUS REGISTER.  BITS 15-10 AND 0
698                             ;ARE DON'T CARE BITS AT THIS TIME AND BIT 7
699                             ;(CONTROLLER READY) IS ALWAYS WRITTEN TO A ONE.
700 014350                      STARS
                                ;;**********************************************************************
701
702
703 014350  012703  002646              MOV     #CSPAT,R3       ;SET UP TABLE POINTER OF PATTERNS
704
705 014354                      BGNSEG                          ;****START OF SEGMENT****
    014354  104004                      EMT     C$BSEG
706
707 014356                      CSTEST:
708 014356  011337  002240              MOV     (R3),GDDAT      ;GET PATTERN INTO GDDAT
709 014362  052737  000200  002240      BIS     #200,GDDAT      ;INSURE GO IS SET
710 014370  013777  002240  165536      MOV     GDDAT,@RLCS     ;LOAD RLCS (CONTROL AND STATUS)
711 014376  032777  040000  165530      BIT     #DERR,@RLCS     ;IF DRIVE ERROR PRESENT
712 014404  001403                      BEQ     99$             ;THEN EXPECT DRIVE AND
713 014406  052737  140000  002240      BIS     #ERR!DERR,GDDAT ;COMPOSITE ERROR
714 014414  017737  165514  002242  99$:MOV     @RLCS,BDDAT     ;READ RLCS BACK
715 014422  042737  000001  002242      BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
716 014430  023737  002240  002242      CMP     GDDAT,BDDAT     ;DID WE READ WHAT WE LOADED
717 014436  001404                      BEQ     1$              ;YES, THEN BRANCH
718
719 014440                              ERRDF   4.,EM5,ERR2     ;WRONG DATA IN RLCS
```

```
        014440  104462                          TRAP    T$ERCODE
        014442  000004                          .WORD   4
        014444  005001                          .WORD   EM5
        014446  010126                          .WORD   ERR2
    720 014450                          1$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
        014450  104010                          EMT     C$ESCAPE
        014452  000012                          .WORD   10000$-.
    721
    722
    723 014454  005723                          TST     (R3)+           ;BUMP FOR NEXT PATTERN
    724 014456  020327  002744                  CMP     R3,#CSEND       ;CHECK FOR END
    725 014462  001335                          BNE     CSTEST          ;NOT END, LOAD NEXT PATTERN
    726
    727 014464                          ENDSEG                          ;****END OF SEGMENT****
        014464                          10000$:
        014464  104005                          EMT     C$ESEG
    728 014466                          ENDTST                          ;****END OF TEST****
        014466                          L10023:
        014466  104001                          EMT     C$ETST
    729
    730
    731                                 .SBTTL  **TEST 6** - READ WRITE OF RLBA
    732
    733 014470                          BGNTST                          ;****START OF TEST****
    734
    735 014470                          STARS
                                        ;;*************************************************************************
    736                                 ;TEST THAT WE CAN WRITE/READ BITS IS THRU 1 OF THE
    737                                 ;BUS ADDRESS REGISTER.  FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
    738                                 ;GROWING 0 AND SHIFTING 0.  BIT 0 IS ALSO LOADED BUT
    739                                 ;SHOULD ALWAYS COME BACK AS 0
    740 014470                          STARS
                                        ;;*************************************************************************
    741
    742
    743 014470  012703  002272                      MOV     #BEGPAT,R3      ;GET START OF PATTERN LIST
    744 014474                          BGNSEG                              ;****START OF SEGMENT****
        014474  104004                          EMT     C$BSEG
    745 014476                          BATEST:
    746 014476  011337  002240                      MOV     (R3),GDDAT      ;GET PATTERN TO SEND
    747 014502  005737  002266                      TST     T.CNTLR         ;RL11??
    748 014506  001403                              BEQ     2$              ;NO,
    749 014510  042737  000001  002240              BIC     #BIT0,GDDAT     ;KEEP RLBA EVEN (UNIBUS)
    750 014516  013777  002240  165412  2$:         MOV     GDDAT,@RLBA     ;LOAD PATTERN TO BUS ADDRESS
    751 014524  017737  165406  002242              MOV     @RLBA,BDDAT     ;READ IT BACK
    752 014532  023737  002240  002242              CMP     GDDAT,BDDAT     ;IS IT CORRECT?
    753 014540  001404                              BEQ     1$              ;IF SO, BRANCH
    754
    755 014542                                      ERRDF   5.,EM6,ERR2     ;DATA WRONG IN RLBA
        014542  104462                              TRAP    T$ERCODE
        014544  000005                              .WORD   5
        014546  005052                              .WORD   EM6
        014550  010126                              .WORD   ERR2
    756 014552                          1$:         ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
        014552  104010                              EMT     C$ESCAPE
        014554  000012                              .WORD   10000$-.
```

```
  014556  005723                          TST     (R3)+            ;BUMP FOR NEXT PATTERN
  014560  020327  002500                  CMP     R3,#ENDPAT       ;CHECK FOR END
  014564  001344                          BNE     BATEST           ;NOT END, BRANCH FOR NEXT

  014566                          ENDSEG                           ;****END OF SEGMENT****
  014566                          10000$:
  014566  104005                          EMT     C$ESEG
  014570                          ENDTST                           ;****END OF TEST****
  014570                          L10024:
  014570  104001                          EMT     C$ETST



                                  .SBTTL  **TEST 7** - READ WRITE OF RLDA

  014572                          BGNTST                           ;****START OF TEST****

  014572                          STARS
                                  ;;*****************************************************************
                                  ;TEST THAT WE CAN WRITE/READ THE DISK ADDRESS REGISTER
                                  ;ALL BIT POSITIONS ARE WRITTEN USING FOUR PATTERNS:
                                  ;GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
  014572                          STARS
                                  ;;*****************************************************************


  014572  012703  002272                  MOV     #BEGPAT,R3       ;SET UP POINTER TO PATTERN LIST
  014576                          BGNSEG                           ;****START OF SEGMENT****
  014576  104004                          EMT     C$BSEG
  014600                          DATEST:
  014600  011337  002240                  MOV     (R3),GDDAT       ;GET PATTERN
  014604  013777  002240  165326          MOV     GDDAT,@RLDA      ;LOAD PATTERN IN DA

  014612  017737  165322  002242          MOV     @RLDA,BDDAT      ;READ PATTERN BACK
  014620  023737  002240  002242          CMP     GDDAT,BDDAT      ;IS IT CORRECT?
  014626  001404                          BEQ     1$               ;BRANCH IF CORRECT

  014630                                  ERRDF   6.,EM7,ERR2      ;WRONG DATA IN RLDA
  014630  104462                          TRAP    T$ERCODE
  014632  000006                          .WORD   6
  014634  005100                          .WORD   EM7
  014636  010126                          .WORD   ERR2
  014640                          1$:     ESCAPE  SEG              ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
  014640  104010                          EMT     C$ESCAPE
  014642  000012                          .WORD   10000$-.


  014644  005723                          TST     (R3)+            ;BUMP POINTER
  014646  020327  002500                  CMP     R3,#ENDPAT       ;AT END OF PATTERNS?
  014652  001352                          BNE     DATEST           ;NO, BRANCH BACK

  014654                          ENDSEG                           ;****END OF SEGMENT****
  014654                          10000$:
  014654  104005                          EMT     C$ESEG
  014656                          ENDTST                           ;****END OF TEST****
  014656                          L10025:
```

```
    014656  104001                          EMT     C$ETST
 42
 43
 44                                 .SBTTL  **TEST 8** - BIS OF RLCS
 45
 46 014660                          BGNTST                          ;****START OF TEST****
 47 014660                          STARS
                                    ;;**********************************************************************
 48                                 ;TEST THAT WE CAN USE THE "BIS" INSTRUCTION ON THE CONTROL
 49                                 ;AND STATUS REGISTER.  BITS 8,9 AND 6-1 ARE TESTED TO
 50                                 ;SET INDIVIDUALLY AS WELL AS COLLECTIVELY WITHOUT DESTROYING
 51                                 ;ANY PREVIOUS DATA PATTERN
 52 014660                          STARS
                                    ;;**********************************************************************
 53
 54
 55 014660  012703  002646                       MOV     #CSPAT,R3       ;GET BEGINNING OF LIST
 56 014664                          BGNSEG                          ;****START OF SEGMENT****
    014664  104004                          EMT     C$BSEG
 57 014666                          1$:
 58 014666  012777  000200  165240            MOV     #CRDY,@RLCS     ;INSURE GO IS THERE
 59 014674  011337  002240                    MOV     (R3),GDDAT      ;SET UP EXPECTED RLCS
 60 014700  052737  000200  002240            BIS     #CRDY,GDDAT     ;IN GDDAT
 61 014706  051377  165222                    BIS     (R3),@RLCS      ;BIT SET PATTERN IN RLCS
 62 014712  032777  040000  165214            BIT     #DERR,@RLCS     ;IF ERROR BIT SET THEN
 63 014720  001403                           BEQ     99$             ;EXPECT IT ON THE READ
 64 014722  052737  140000  002240            BIS     #ERR!DERR,GDDAT ;BACK
 65 014730  017737  165200  002242  99$:      MOV     @RLCS,BDDAT     ;READ RLCS TO CHECK "BIS"
 66 014736  042737  000001  002242            BIC     #DRDY,BDDAT     ;CLEAR OUT DRIVE READY
 67 014744  023737  002242  002240            CMP     BDDAT,GDDAT     ;DID BIS WORK?
 68 014752  001404                           BEQ     2$              ;BRANCH IF OKAY
 69
 70 014754                                  ERRDF   7.,EM61,ERR2    ;WRONG DATA IN RLCS
    014754  104462                          TRAP    T$ERCODE
    014756  000007                          .WORD   7
    014760  006575                          .WORD   EM61
    014762  010126                          .WORD   ERR2
 71 014764                          2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
    014764  104010                          EMT     C$ESCAPE
    014766  000012                          .WORD   10000$-.
 72                                                                 ;BIT OR CLEARED OTHER BIT
 73
 74 014770  005723                          TST     (R3)+           ;GET NEXT PATTERN
 75 014772  022703  002744                    CMP     #CSEND,R3       ;AT END OF LIST
 76 014776  001333                           BNE     1$              ;NO GO BACK FOR TEST OF
 77                                                                 ;NEXT PATTERN
 78 015000                          ENDSEG                          ;****END OF SEGMENT****
    015000                          10000$:
    015000  104005                          EMT     C$ESEG
 79 015002                          ENDTST                          ;****END OF TEST****
    015002                          L10026:
    015002  104001                          EMT     C$ETST
 80
 81
 82                                 .SBTTL  **TEST 9** - BIC OF RLCS
 83
 84 015004                          BGNTST                          ;****START OF TEST****
```

```
     85
     86 015004                          STARS
                                        ;;*****************************************************************
     87                                 ;TEST THAT THE "BIC" INSTRUCTION WILL WORK ON THE
     88                                 ;CONTROL AND STATUS REGISTER.  BITS 8-9 AND 6-1 ARE
     89                                 ;TESTED.
     90 015004                          STARS
                                        ;;*****************************************************************
     91
     92
     93 015004  012703  002646                  MOV     #CSPAT,R3       ;GET BEGINNING OF PATTERNS
     94 015010                          BGNSEG                          ;****START OF SEGMENT****
        015010  104004                          EMT     C$BSEG
     95 015012                          1$:
     96 015012  012777  001776  165114          MOV     #1776,@RLCS     ;SET ALL SETTABLE BITS
     97 015020  012737  001776  002240          MOV     #1776,GDDAT     ;SET UP EXPECT DATA IN
     98 015026  041337  002240                  BIC     (R3),GDDAT      ;GDDAT
     99 015032  041377  165076                  BIC     (R3),@RLCS      ;CLEAR BITS IN RLCS VIA "BIC"
    100 015036  032777  040000  165070          BIT     #DERR,@RLCS     ;IF DRIVE ERROR BIT SET
    101 015044  001403                          BEQ     99$             ;EXPECT IT SET WHEN WE
    102 015046  052737  140000  002240          BIS     #ERR!DERR,GDDAT ;READ IT BACK
    103 015054  017737  165054  002242  99$:     MOV     @RLCS,BDDAT     ;MOVE RLCS TO BDDAT FOR COMPARE
    104 015062  042737  000001  002242          BIC     #DRDY,BDDAT     ;CLEAR DRIVE READY
    105 015070  023737  002242  002240          CMP     BDDAT,GDDAT     ;DID "BIC" WORK PROPERLY
    106 015076  001404                          BEQ     2$              ;BRANCH IF OKAY
    107
    108 015100                                  ERRDF   8.,EM62,ERR2    ;WRONG DATA IN RLCS
        015100  104462                          TRAP    T$ERCODE
        015102  000010                          .WORD   8
        015104  006656                          .WORD   EM62
        015106  010126                          .WORD   ERR2
    109 015110                          2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
        015110  104010                          EMT     C$ESCAPE
        015112  000012                          .WORD   10000$-.
    110
    111 015114  005723                          TST     (R3)+           ;GET NEXT PATTERN
    112 015116  020327  002744                  CMP     R3,#CSEND       ;AT END OF LIST
    113 015122  001333                          BNE     1$              ;NO, GO BACK WITH NEXT PATTERN
    114 015124                          ENDSEG                          ;****END OF SEGMENT****
        015124                          10000$:
        015124  104005                          EMT     C$ESEG
    115 015126                          ENDTST                          ;****END OF TEST****
        015126                          L10027:
        015126  104001                          EMT     C$ETST
    116
    117
    118                                 .SBTTL  **TEST 10** - BIS OF RLBA
    119
    120 015130                          BGNTST                          ;****START OF TEST****
    121
    122 015130                          STARS
                                        ;;*****************************************************************
    123                                 ;TEST THAT THE "BIS" INSTRUCTION WILL WORK ON THE BUS
    124                                 ;ADDRESS REGISTER.  BITS 15-0 ARE LOADED, ONLY BITS 15-1
    125                                 ;ARE EXPECTED BACK.  FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
    126                                 ;GROWING 0, AND SHIFTING 0.
    127 015130                          STARS
```

```
                                   ;;*******************************************************************
    015130  012703  002272                     MOV     #BEGPAT,R3          ;GET START OF LIST
    015134                         BGNSEG                                  ;****START OF SEGMENT****
    015134  104004                             EMT     C$BSEG
    015136                         1$:
    015136  005077  164774                     CLR     @RLBA               ;CLEAR "BA"
    015142  011337  002240                     MOV     (R3),GDDAT          ;SET EXPECTED
    015146  005737  002266                     TST     T.CNTLR             ;RL11
    015152  001403                             BEQ     3$                  ;NO
    015154  042737  000001  002240             BIC     #1,GDDAT            ;BIT 0 CAN'T SET IN RLBA (UNIBUS)
    015162  051377  164750         3$:         BIS     (R3),@RLBA          ;BIS RLBA WITH PATTERN
    015166  017737  164744  002242             MOV     @RLBA,BDDAT         ;READ "BA"
    015174  023737  002242  002240             CMP     BDDAT,GDDAT         ;DID RLBA LOAD PROPERLY?
    015202  001404                             BEQ     2$                  ;BRANCH IF YES

    015204                                     ERRDF   9.,EM63,ERR2        ;WRONG DATA IN RLBA
    015204  104462                             TRAP    T$ERCODE
    015206  000011                             .WORD   9
    015210  006741                             .WORD   EM63
    015212  010126                             .WORD   ERR2
    015214                         2$:         ESCAPE  SEG                 ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
    015214  104010                             EMT     C$ESCAPE
    015216  000012                             .WORD   10000$-.

    015220  005723                             TST     (R3)+               ;GET NEXT PATTERN
    015222  020327  002500                     CMP     R3,#ENDPAT          ;DID WE COMPLETE LIST
    015226  001343                             BNE     1$                  ;NO, GO BACK FOR NEXT.
    015230                         ENDSEG                                  ;****END OF SEGMENT****
    015230                         10000$:
    015230  104005                             EMT     C$ESEG
    015232                         ENDTST                                  ;****END OF TEST****
    015232                         L10030:
    015232  104001                             EMT     C$ETST



                                   .SBTTL  **TEST 11** - BIC OF RLBA

    015234                         BGNTST                                  ;****START OF TEST****

    015234                         STARS
                                   ;;*******************************************************************
                                   ;TEST THAT THE "BIC" INSTRUCTION WILL WORK ON THE BUS
                                   ;ADDRESS REGISTER.  BITS 15-1 ARE TESTED WITH 4 PATTERNS
                                   ;GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0.
    015234                         STARS
                                   ;;*******************************************************************


    015234  012703  002272                     MOV     #BEGPAT,R3          ;GET START OF LIST
    015240                         BGNSEG                                  ;****START OF SEGMENT****
    015240  104004                             EMT     C$BSEG
    015242                         1$:
    015242  012777  177776  164666             MOV     #-2,@RLBA           ;SET RLBA TO ALL 1'S (BIT 0=0)
    015250  012737  177776  002240             MOV     #-2,GDDAT           ;SET UP EXPECTED RESULTS
    015256  041337  002240                     BIC     (R3),GDDAT          ;IN GDDAT
```

```
    170 015262  041377  164650                  BIC     (R3),@RLBA      ;BIC RLBA
    171 015266  017737  164644  002242          MOV     @RLBA,BDDAT     ;READ RLBA
    172 015274  023737  002242  002240          CMP     BDDAT,GDDAT     ;BIC WORK OKAY?
    173 015302  001404                          BEQ     2$              ;IF YES BRANCH
    174
    175 015304                                  ERRDF   10.,EM64,ERR2   ;WRONG DATA IN RLBA
        015304  104462                          TRAP    T$ERCODE
        015306  000012                          .WORD   10
        015310  007022                          .WORD   EM64
        015312  010126                          .WORD   ERR2
    176 015314                          2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
        015314  104010                          EMT     C$ESCAPE
        015316  000012                          .WORD   10000$-.
    177
    178 015320  005723                          TST     (R3)+           ;GET NEXT PATTERN
    179 015322  020327  002500                  CMP     R3,#ENDPAT      ;HAVE WE COMPLETED LIST
    180 015326  001345                          BNE     1$              ;NO, GO BACK FOR NEXT
    181 015330                          ENDSEG                          ;****END OF SEGMENT****
        015330                          10000$:
        015330  104005                          EMT     C$ESEG
    182 015332                          ENDTST                          ;****END OF TEST****
        015332                          L10031:
        015332  104001                          EMT     C$ETST
    183
    184
    185                                 .SBTTL  **TEST 12** - BIS OF RLDA
    186
    187 015334                          BGNTST                          ;****START OF TEST****
    188
    189 015334                          STARS
                                        ;;**************************************************************
    190                                 ;TEST THAT THE "BIS" INSTRUCTION WILL WORK ON THE DISK ADDRESS
    191                                 ;REGISTER.  BITS 15-0 ARE TESTED WITH 4 PATTERNS, GROWING 1,
    192                                 ;SHIFTING 1, GROWING 0, AND SHIFTING 0.
    193 015334                          STARS
                                        ;;**************************************************************
    194
    195
    196 015334  012703  002272                  MOV     #BEGPAT,R3      ;GET START OF LIST
    197 015340                          BGNSEG                          ;****START OF SEGMENT****
        015340  104004                          EMT     C$BSEG
    198 015342                          1$:
    199 015342  005077  164572                  CLR     @RLDA           ;CLEAR "DA"
    200 015346  011337  002240                  MOV     (R3),GDDAT      ;SET EXPECTED
    201 015352  051377  164562                  BIS     (R3),@RLDA      ;BIS RLDA
    202 015356  017737  164556  002242          MOV     @RLDA,BDDAT     ;READ RLDA
    203 015364  023737  002242  002240          CMP     BDDAT,GDDAT     ;IS RLDA CORRECT
    204 015372  001404                          BEQ     2$              ;IF OKAY BRANCH
    205
    206 015374                                  ERRDF   11.,EM65,ERR2   ;WRONG DATA IN RLDA
        015374  104462                          TRAP    T$ERCODE
        015376  000013                          .WORD   11
        015400  007105                          .WORD   EM65
        015402  010126                          .WORD   ERR2
    207 015404                          2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
        015404  104010                          EMT     C$ESCAPE
        015406  000012                          .WORD   10000$-.
```

```
015410  005723                           TST     (R3)+           ;GET NEXT PATTERN
015412  020327  002500                   CMP     R3,#ENDPAT      ;HAVE WE FINISHED?
015416  001351                           BNE     1$              ;NO GO BACK
015420                           ENDSEG                          ;****END OF SEGMENT****
015420                           10000$:
015420  104005                           EMT     C$ESEG
015422                           ENDTST                          ;****END OF TEST****
015422                           L10032:
015422  104001                           EMT     C$ETST


                                 .SBTTL  **TEST 13** - BIC OF RLDA

015424                           BGNTST                          ;****START OF TEST****

015424                           STARS
                                 ;;*****************************************************************
                                 ;TEST THAT THE "BIC" INSTRUCTION WORKS ON THE DISK
                                 ;ADDRESS REGISTER.  ALL BITS ARE TESTED WITH FOUR
                                 ;PATTERNS: GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
015424                           STARS
                                 ;;*****************************************************************


015424  012703  002272                   MOV     #BEGPAT,R3      ;GET START OF LIST
015430                           BGNSEG                          ;****START OF SEGMENT****
015430  104004                           EMT     C$BSEG
015432                           1$:
015432  012777  177777  164500           MOV     #-1,@RLDA       ;SET RLDA TO ALL 1'S
015440  012737  177777  002240           MOV     #-1,GDDAT       ;SET EXPECTED DATA
015446  041337  002240                   BIC     (R3),GDDAT      ;SET EXPECTED DATA
015452  041377  164462                   BIC     (R3),@RLDA      ;"BIC" RLDA
015456  017737  164456  002242           MOV     @RLDA,BDDAT     ;READ RLDA
015464  023737  002240  002242           CMP     GDDAT,BDDAT     ;DID "BIC" WORK?
015472  001404                           BEQ     2$              ;IF IT DID BRANCH

015474                                   ERRDF   12.,EM66,ERR2   ;WRONG DATA IN RLDA
015474  104462                           TRAP    T$ERCODE
015476  000014                           .WORD   12
015500  007166                           .WORD   EM66
015502  010126                           .WORD   ERR2
015504                           2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
015504  104010                           EMT     C$ESCAPE
015506  000012                           .WORD   10000$-.

015510  005723                           TST     (R3)+           ;GET NEXT PATTERN
015512  020327  002500                   CMP     R3,#ENDPAT      ;DONE?
015516  001345                           BNE     1$              ;NO GO BACK
015520                           ENDSEG                          ;****END OF SEGMENT****
015520                           10000$:
015520  104005                           EMT     C$ESEG
015522                           ENDTST                          ;****END OF TEST****
015522                           L10033:
015522  104001                           EMT     C$ETST
```

```
                                .SBTTL  **TEST 14** - BUS RESET OF RLCS

                                BGNTST                          ;****START OF TEST****

                                STARS
                                ;;*****************************************************************
                                ;TEST THAT A BUS RESET WILL CLEAR THE PROPER BITS
                                ;OF THE CONTROL AND STATUS REGISTER.  THOSE BITS ARE
                                ;BITS 6-1,8,9,10,11,12,13,15.  BIT 15 WILL CLEAR ONLY
                                ;IF BIT 14 (DRIVE ERROR IS NOT SET).  BIT 0 (DRIVE READY)
                                ;IS A DON'T CARE.  IF AT THE START UP THIS TEST BIT
                                ;14 (DRIVE ERROR) IS SET WE WILL INSIST IF IS THERE AFTER
                                ;THE "RESET" ALONG WITH BIT 15 (COMPOSITE ERROR).  BITS
                                ;15-10 ARE NOT WRITEABLE.
                                STARS
                                ;;*****************************************************************

015524                                  SETPRI  #PRI07          ;PRIORITY TO SEVEN
015524  012700  000340                  MOV     #PRI07,R0
015530  104041                          EMT     C$SPRI
015532  012777  000377  164374          MOV     #377,@RLCS      ;LOAD ALL RLCS LOADABLE BITS
015540  012737  000200  002240          MOV     #CRDY,GDDAT     ;SETUP EXPECTED
015546  032777  040000  164360          BIT     #DERR,@RLCS     ;DRIVE ERR SET?
015554  001403                          BEQ     1$              ;IF NOT DON'T EXPECT IT
015556  052737  140000  002240          BIS     #DERR!ERR,GDDAT  ;IT'S SET, INIT BETTER NOT CLR
015564  012700  000100          1$:     MOV     #100,R0         ;SET UP A WAIT LOOP
015570                                  BRESET                  ;BUS RESET
015570  104033                          EMT     C$RESET
015572  005300                  2$:     DEC     R0              ;WAIT IN CASE OF DRIVE ERROR
015574  001376                          BNE     2$              ;
015576  017737  164332  002242          MOV     @RLCS,BDDAT     ;READ RLCS
015604  042737  000001  002242          BIC     #DRDY,BDDAT     ;CLEAR OUT DRDY - DON'T CARE
015612  023737  002242  002240          CMP     BDDAT,GDDAT     ;DID INIT WORK
015620  001404                          BEQ     3$              ;YES, BRANCH

015622                                  ERRDF   13.,EM67,ERR2   ;WRONG DATA IN RLCS
015622  104462                          TRAP    T$ERCODE
015624  000015                          .WORD   13
015626  007251                          .WORD   EM67
015630  010126                          .WORD   ERR2
015632                          3$:
015632                          ENDTST                          ;****END OF TEST****
015632                          L10034:
015632  104001                          EMT     C$ETST

                                .SBTTL  **TEST 15** - BUS RESET OF RLBA

015634                          BGNTST                          ;****START OF TEST****

015634                          STARS
                                ;;*****************************************************************
                                ;TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
                                ;BUS ADDRESS REGISTER.  THE BUS ADDRESS IS LOADED WITH 177776
                                ;AND IS EXPECTED TO BE ZERO AFTER THE RESET
015634                          STARS
```

```
                                ;;*****************************************************************
293
294
295 015634  012777  177776  164274        MOV     #-2,@RLBA       ;SET BA TO ALL 1'S
296 015642  005737  002266                TST     T.CNTLR         ;RL11??
297 015646  001403                        BEQ     2$              ;NO
298 015650  052777  000001  164260        BIS     #1,@RLBA
299 015656  005037  002240          2$:   CLR     GDDAT           ;CLEAR EXPECTED DATA
300 015662                                BRESET                  ;ISSUE BUS INIT
    015662  104033                        EMT     C$RESET
301 015664  017737  164246  002242        MOV     @RLBA,BDDAT     ;READ RLBA
302 015672  001404                        BEQ     1$              ;IF CLEAR BRANCH
303
304 015674                                ERRDF   14.,EM70,ERR2   ;WRONG DATA IN RLBA
    015674  104462                        TRAP    T$ERCODE
    015676  000016                        .WORD   14
    015700  007306                        .WORD   EM70
    015702  010126                        .WORD   ERR2
305 015704                          1$:
306
307 015704                          ENDTST                      ;****END OF TEST****
    015704                          L10035:
    015704  104001                        EMT     C$ETST
308
309
310                                 .SBTTL  **TEST 16** - BUS RESET OF RLDA
311
312 015706                          BGNTST                      ;****START OF TEST****
313
314 015706                          STARS
                                    ;;*****************************************************************
315                                 ;TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
316                                 ;DISK ADDRESS REGISTER.  THE DISK ADDRESS IS LOADED WITH 177777
317                                 ;AND IS EXPECTED TO BE ZERO AFTER THE RESET.
318 015706                          STARS
                                    ;;*****************************************************************
319
320
321 015706  012777  177777  164224        MOV     #-1,@RLDA       ;SET DA TO ALL 1'S
322 015714  005037  002240                CLR     GDDAT           ;CLEAR EXPECTED
323 015720                                BRESET                  ;ISSUE BUS INIT
    015720  104033                        EMT     C$RESET
324 015722  017737  164212  002242        MOV     @RLDA,BDDAT     ;READ RLDA
325 015730  001404                        BEQ     1$              ;IF CLEAR BRANCH
326
327 015732                                ERRDF   15.,EM71,ERR2   ;WRONG DATA IN RLDA
    015732  104462                        TRAP    T$ERCODE
    015734  000017                        .WORD   15
    015736  007343                        .WORD   EM71
    015740  010126                        .WORD   ERR2
328 015742                          1$:
329
330 015742                          ENDTST                      ;****END OF TEST****
    015742                          L10036:
    015742  104001                        EMT     C$ETST
331
332
```

```
                                .SBTTL  **TEST 17** - UNIQUENESS OF RLCS

015744                          BGNTST                          ;****START OF TEST****

015744                          STARS
                                ;;*****************************************************************
                                ;TEST THE UNIQUENESS OF THE CONTROL AND STATUS
                                ;REGISTER.  THE RLBA AND RLDA ARE PRELOADED WITH
                                ;177776 AND 177777 RESPECTIVELY.  THE RLCS IS THEN
                                ;LOADED TO INSURE THAT NEITHER THE RLBA OR RLDA
                                ;ARE MODIFIED BY THE WRITING OF THE RLCS.
015744                          STARS
                                ;;*****************************************************************

015744  012737  000201  002210          MOV     #DRDY!CRDY,LDCSR        ;SET DRIVE AND CONTROLLER READY
015752  012777  177776  164156          MOV     #-2,@RLBA               ;SET RLBA TO ALL 1'S
015760  012777  177777  164152          MOV     #-1,@RLDA               ;SET RLDA TO ALL 1'S
015766  013777  002210  164140          MOV     LDCSR,@RLCS             ;WRITE RLCS

                                ;CHECK THAT RLBA REMAINED UNEFFECTED

015774  022777  177776  164134          CMP     #-2,@RLBA       ;RLBA OKAY?
016002  001412                        BEQ     1$              ;YES, GO CHECK DA

016004  012737  177776  002240          MOV     #-2,GDDAT       ;SET UP EXPECTED
016012  017737  164120  002242          MOV     @RLBA,BDDAT     ;READ RLBA

016020                                  ERRDF   16.,EM72,ERR2   ;CS MODIFIED BA
016020  104462                        TRAP    T$ERCODE
016022  000020                        .WORD   16
016024  007400                        .WORD   EM72
016026  010126                        .WORD   ERR2
016030                          1$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
016030  104006                        EMT     C$CLP1

016032  022777  177777  164100          CMP     #-1,@RLDA       ;RLDA OKAY?
016040  001412                        BEQ     2$              ;YES, CONTINUE

016042  012737  177777  002240          MOV     #-1,GDDAT       ;SET UP EXPECTED
016050  017737  164064  002242          MOV     @RLDA,BDDAT     ;READ DA

016056                                  ERRDF   17.,EM73,ERR2   ;CS MODIFIED DA
016056  104462                        TRAP    T$ERCODE
016060  000021                        .WORD   17
016062  007433                        .WORD   EM73
016064  010126                        .WORD   ERR2
016066                          2$:

016066                          ENDTST                          ;****END OF TEST****
016066                          L10037:
016066  104001                        EMT     C$ETST

                                .SBTTL  **TEST 18** - UNIQUENESS OF RLBA
```

```
                                BGNTST                          ;****START OF TEST****
                                STARS
                                ;;*****************************************************************
                                ;TEST THE UNIQUENESS OF THE BUS ADDRESS REGISTER.  THE
                                ;RLCS AND RLDA ARE LOADED WITH XXX20X AND 177777
                                ;RESPECTIVELY.  THE RLBA IS THEN WRITTEN TO INSURE
                                ;THAT NEITHER THE RLCS OR RLDA ARE MODIFIED
                                ;BY WRITING THE RLBA.
                                STARS
                                ;;*****************************************************************

016070  012737  000200  002240          MOV     #CRDY,GDDAT     ;CONTROLLER READY
016076  032777  040000  164030          BIT     #DERR,@RLCS     ;IF DRIVE ERROR IS
016104  001403                          BEQ     99$             ;SET THEN EXPECT IT
016106  052737  140000  002240          BIS     #ERR!DERR,GDDAT ;SET WHEN WE READ IT.
016114  013777  002240  164012  99$:    MOV     GDDAT,@RLCS     ;LOAD RLCS
016122  012777  177777  164010          MOV     #-1,@RLDA       ;LOAD RLDA
016130  005077  164002                  CLR     @RLBA           ;CLEAR RLBA

                                ;CHECK IF RLCS IS OKAY

016134  017737  163774  002242          MOV     @RLCS,BDDAT     ;READ RLCS
016142  042737  000001  002242          BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
016150  023737  002242  002240          CMP     BDDAT,GDDAT     ;CS OK?
016156  001404                          BEQ     1$              ;YES, GO CHECK DA

016160                                  ERRDF   18.,EM74,ERR2   ;BA MODIFIED CS
016160  104462                          TRAP    T$ERCODE
016162  000022                          .WORD   18
016164  007466                          .WORD   EM74
016166  010126                          .WORD   ERR2
016170                          1$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
016170  104006                          EMT     C$CLP1

016172  022777  177777  163740          CMP     #-1,@RLDA       ;IS RLDA OKAY?

016200  001412                          BEQ     2$              ;IF OKAY BRANCH

016202  012737  177777  002240          MOV     #-1,GDDAT       ;SET UP EXPECTED
016210  017737  163724  002242          MOV     @RLDA,BDDAT     ;READ RLDA

016216                                  ERRDF   19.,EM75,ERR2   ;BA MODIFIED DA
016216  104462                          TRAP    T$ERCODE
016220  000023                          .WORD   19
016222  007520                          .WORD   EM75
016224  010126                          .WORD   ERR2
016226                          2$:
016226                          ENDTST                          ;****END OF TEST****
016226                          L10040:
016226  104001                          EMT     C$ETST


                                .SBTTL  **TEST 19** - UNIQUENESS OF RLDA

016230                          BGNTST                          ;****START OF TEST****
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;TEST THE UNIQUENESS OF THE DISK ADDRESS REGISTER.  THE RLCS
                                        ;AND RLBA ARE LOADED WITH XXX20X AND 177776
                                        ;RESPECTIVELY.  THE RLDA IS THEN WRITTEN TO INSURE
                                        ;THAT NEITHER THE RLCS OR THE RLBA ARE MODIFIED
                                        ;BY WRITING THE RLDA.
                                        STARS
                                        ;;*****************************************************************

016230  012737  000200  002240          MOV     #CRDY,GDDAT     ;CONTROLLER READY
016236  032777  040000  163670          BIT     #DERR,@RLCS     ;IF DRIVE ERROR SET
016244  001403                          BEQ     99$             ;THEN EXPECT IT LATER
016246  052737  140000  002240          BIS     #ERR!DERR,GDDAT
016254  013777  002240  163652  99$:    MOV     GDDAT,@RLCS     ;LOAD CS
016262  012777  177776  163646          MOV     #-2,@RLBA       ;LOAD BA WITH ALL 1'S
016270  005077  163644                  CLR     @RLDA           ;CLEAR RLDA

                                        ;CHECK IF RLCS IS OKAY

016274  017737  163634  002242          MOV     @RLCS,BDDAT     ;READ RLCS
016302  042737  000001  002242          BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
016310  023737  002240  002242          CMP     GDDAT,BDDAT     ;RLCS OKAY?
016316  001404                          BEQ     1$              ;YES, THEN BRANCH

016320                                  ERRDF   20.,EM76,ERR2   ;DA MODIFIED CS
016320  104462                          TRAP    T$ERCODE
016322  000024                          .WORD   20
016324  007552                          .WORD   EM76
016326  010126                          .WORD   ERR2
016330                          1$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
016330  104006                          EMT     C$CLP1

016332  022777  177776  163576          CMP     #-2,@RLBA       ;IS RLBA OKAY?
016340  001412                          BEQ     2$              ;BRANCH IF OKAY

016342  012737  177776  002240          MOV     #-2,GDDAT       ;SET UP EXPECTED
016350  017737  163562  002242          MOV     @RLBA,BDDAT     ;READ RLBA

016356                                  ERRDF   21.,EM77,ERR2   ;DA MODIFIED BA
016356  104462                          TRAP    T$ERCODE
016360  000025                          .WORD   21
016362  007605                          .WORD   EM77
016364  010126                          .WORD   ERR2
016366                          2$:

                                        ENDTST                  ;****END OF TEST****
016366                                  L10041:
016366  104001                          EMT     C$ETST

                                        .SBTTL  **TEST 20** - UNIQUENESS OF RLMP

016370                                  BGNTST                  ;****START OF TEST****
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;TEST THE UNIQUENESS OF THE MULTI-PURPOSE REGISTER
                                        ;WE WILL WRITE THE RLCS, RLBA, AND THE RLDA, THEN THE
                                        ;RLMP IS WRITTEN. WE THEN GO BACK AN VERIFY THE CONTENTS
                                        ;OF THE RLCS, RLBA, RLDA.
                                        STARS
                                        ;;*****************************************************************

016370  012737  000200  002240          MOV     #CRDY,GDDAT     ;CONTROLLER READY
016376  032777  040000  163530          BIT     #DERR,@RLCS     ;IF DRIVE ERROR SET
016404  001403                          BEQ     99$             ;THE EXPECT IT LATER
016406  052737  140000  002240          BIS     #ERR!DERR,GDDAT
016414  013777  002240  163512  99$:    MOV     GDDAT,@RLCS     ;LOAD CS
016422  012777  177776  163506          MOV     #-2,@RLBA       ;LOAD BA WITH ALL 1'S
016430  012777  177777  163502          MOV     #-1,@RLDA       ;LOAD RLDA
016436  005077  163500                  CLR     @RLMP           ;WRITE RLMP

                                        ;CHECK IF RLCS IS OKAY

016442  017737  163466  002242          MOV     @RLCS,BDDAT     ;READ RLCS
016450  042737  000001  002242          BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
016456  023737  002240  002242          CMP     GDDAT,BDDAT     ;RLCS OKAY?
016464  001404                          BEQ     1$              ;YES, THEN BRANCH

016466                                  ERRDF   201.,EM44,ERR2  ;MP MODIFIED CS
016466  104462                          TRAP    T$ERCODE
016470  000311                          .WORD   201
016472  006121                          .WORD   EM44
016474  010126                          .WORD   ERR2
016476                          1$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
016476  104006                          EMT     C$CLP1

016500  022777  177776  163430          CMP     #-2,@RLBA       ;IS RLBA OKAY?
016506  001412                          BEQ     2$              ;BRANCH IF OKAY

016510  012737  177776  002240          MOV     #-2,GDDAT       ;SET UP EXPECTED
016516  017737  163414  002242          MOV     @RLBA,BDDAT     ;READ RLBA

016524                                  ERRDF   211.,EM45,ERR2  ;MP MODIFIED BA
016524  104462                          TRAP    T$ERCODE
016526  000323                          .WORD   211
016530  006154                          .WORD   EM45
016532  010126                          .WORD   ERR2
016534                          2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
016534  104006                          EMT     C$CLP1
016536  022777  177777  163374          CMP     #-1,@RLDA       ;DISK ADDRESS OKAY
016544  001412                          BEQ     3$              ;YES, CONTINUE

016546  017737  163366  002242          MOV     @RLDA,BDDAT     ;SET UP BAD
016554  012737  177777  002240          MOV     #-1,GDDAT       ;SET UP EXPECTED

016562                                  ERRDF   212.,EM46,ERR2  ;MP MODIFIED DA
016562  104462                          TRAP    T$ERCODE
016564  000324                          .WORD   212
```

```
        016566  006207                          .WORD   EM46
        016570  010126                          .WORD   ERR2
    508
    509 016572                          3$:
    510
    511
    512 016572                          ENDTST                          ;****END OF TEST****
        016572                          L10042:
        016572  104001                          EMT     C$ETST
    513
    514                                 .SBTTL  **TEST 21** - NOOP FUNCTION(RL11 ONLY)
    515
    516 016574                          BGNTST                          ;****START OF TEST****
    517
    518
    519
    520 016574                          STARS
                                        ;;**************************************************************
    521                                 ;TEST THAT NOOP WILL FUNCTION.  WE WILL ISSUE THE
    522                                 ;NOOP AND WAIT FOR CONTROLLER READY TO SET.  A
    523                                 ;TIMEOUT OF 200 MILLISECS IS ALLOWED.  DRIVE 0 IS ALWAYS
    524                                 ;SELECTED SINCE THE DRIVE IS NOT NECESSARY.
    525 016574                          STARS
                                        ;;**************************************************************
    526
    527
    528 016574  005737  002266                  TST     T.CNTLR         ;RLV11??
    529 016600  001410                          BEQ     99$             ;YES SKIP TEST
    530
    531
    532 016602  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
    533 016606  000000                          NOOP0                   ;NOOP(0) FUNCTION
    534 016610  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
    535 016614                          2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
        016614  104006                          EMT     C$CLP1
    536
    537 016616  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
    538
    539 016622                          99$:
    540 016622                          ENDTST                          ;****END OF TEST****
        016622                          L10043:
        016622  104001                          EMT     C$ETST
    541
    542
    543                                 .SBTTL  **TEST 22** - TEST NOOP DOES NOTHING
    544
    545 016624                          BGNTST                          ;****START OF TEST****
    546
    547 016624                          STARS
                                        ;;**************************************************************
    548                                 ;TEST THAT ISSUING A NOOP FUNCTION DOES NOTHING.  THIS IS DONE BY WRITING
    549                                 ;THE RLBA, AND RLDA, READING THE RLMP AND MAKING SURE NOTHING CHANGES.
    550 016624                          STARS
                                        ;;**************************************************************
    551
    552 016624  005737  002266                  TST     T.CNTLR         ;RLV11??
    553 016630  001476                          BEQ     3$
```

```
554
555 016632  012777  000001  163300          MOV     #1,@RLDA        ;LOAD DISK ADDRESS
556 016640  012777  000002  163270          MOV     #2,@RLBA        ;LOAD BUS ADDRESS
557 016646  005077  163270                  CLR     @RLMP
558 016652  017737  163264  002240          MOV     @RLMP,GDDAT     ;READ RLMP
559
560 016660  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
561 016664  000000                          NOOP0
562 016666  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
563 016672                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
    016672  104006                          EMT     C$CLP1
564
565 016674  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
566 016700                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
    016700  104010                          EMT     C$ESCAPE
    016702  000124                          .WORD   L10044-.
567
568 016704  017737  163232  002242          MOV     @RLMP,BDDAT     ;READ RLMP
569 016712  023737  002240  002242          CMP     GDDAT,BDDAT     ;RLMP OK?
570 016720  001404                          BEQ     1$
571
572 016722                                  ERRDF   202.,EM14,ERR2
    016722  104462                          TRAP    T$ERCODE
    016724  000312                          .WORD   202
    016726  005221                          .WORD   EM14
    016730  010126                          .WORD   ERR2
573
574 016732                          1$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
    016732  104006                          EMT     C$CLP1
575
576 016734  012737  000002  002240          MOV     #2,GDDAT        ;SET UP EXP'D BA
577 016742  017737  163170  002242          MOV     @RLBA,BDDAT     ;READ BA
578 016750  023737  002240  002242          CMP     GDDAT,BDDAT     ;BA OK?
579 016756  001404                          BEQ     2$              ;YES
580
581 016760                                  ERRDF   203.,EM15,ERR2
    016760  104462                          TRAP    T$ERCODE
    016762  000313                          .WORD   203
    016764  005247                          .WORD   EM15
    016766  010126                          .WORD   ERR2
582
583 016770                          2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
    016770  104006                          EMT     C$CLP1
584
585 016772  012737  000001  002240          MOV     #1,GDDAT        ;SET UP EXP'D DA
586 017000  017737  163134  002242          MOV     @RLDA,BDDAT     ;READ DA
587 017006  023737  002240  002242          CMP     GDDAT,BDDAT     ;DA OKAY
588 017014  001404                          BEQ     3$
589
590 017016                                  ERRDF   204.,EM16,ERR2
    017016  104462                          TRAP    T$ERCODE
    017020  000314                          .WORD   204
    017022  005275                          .WORD   EM16
    017024  010126                          .WORD   ERR2
591
592 017026                          3$:
593
```

```
                                ENDTST                          ;****END OF TEST****
                                L10044:
017026  104001                          EMT     C$ETST

                                .SBTTL  **1EST 23** - TEST OF INTERRUPT

                                BGNTST                          ;****START OF TEST****

                                STARS
                                ;;**************************************************************
                                ;CHECK THE INTERRUPT WITH A NOOP.  WE WILL SET UP THE
                                ;INTERRUPT VECTOR, LOWER THE PSW TO ZERO AND ISSUE
                                ;A NOOP.  THE INTERRUPT SERVICE ROUTINE WILL SET A
                                ;FLAG UPON INTERRUPT AND RETURN IN LINE.  WE WAIT 200 MILLISECONDS
                                ;LOOKING FOR THAT FLAG TO BE SET BEFORE CALLING IT
                                ;AN ERROR.  IF THE INTERRUPT SENDS US TO ANOTHER
                                ;VECTOR ADDRESS THEN THE ERROR HANDLER WILL REPORT
                                ;"TRAP TO XXXX FROM YYYY" AND RETURN  TO DIAG SUP MONITOR.  IF THE
                                ;INTERRUPT GOES TO ABOVE 1000 WHO KNOWS WHAT WILL HAPPEN.
                                STARS
                                ;;**************************************************************

017030  005737  002266                TST     T.CNTLR
017034  001426                        BEQ     99$

017036  005037  002206                CLR     INTFLG          ;CLEAR INTERRUPT OCCRUANCE FLAG
017042                          SETPRI  #PR100          ;SET PSW TO 0
017042  012700  000000                MOV     #PR100,R0
017046  104041                        EMT     C$SPRI
017050  004537  013102                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
017054  000100                        NOOP0!INTEN             ;NOOP AND INTERRUPT ENABLE
017056  004537  013712                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
017062  005737  002206                TST     INTFLG          ;DID INTERRUPT OCCUR
017066  001004                        BNE     2$              ;IF SO BRANCH
017070                          ERRDF   22.,EM13,ERR0
017070  104462                        TRAP    T$ERCODE
017072  000026                        .WORD   22
017074  005167                        .WORD   EM13
017076  010076                        .WORD   ERR0
017100  005037  002206        2$:     CLR     INTFLG
017104                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
017104  104006                        EMT     C$CLP1
017106  004537  012602                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS

017112                          99$:
017112                          ENDTST                          ;****END OF TEST****
017112                          L10045:
017112  104001                        EMT     C$ETST

                                .SBTTL  **TEST 24** - TEST PRIORITY BR LEVEL

017114                          BGNTST                          ;****START OF TEST****
```

```
                                        STARS
                                        ;;*****************************************************************
                                        ;TEST THAT PRIORITY GIVEN IS ACTUAL PRIORITY OF CONTROLLER.  WE KNOW
                                        ;THE BOARD WILL INTERRUPT.  WE WILL START TRYING TO INTERRUPT AT 7
                                        ;AND WORK DOWN TIL IT DOES INTERRUPT.
                                        STARS
                                        ;;*****************************************************************

017114  005737  002266                  TST     T.CNTLR         ;RLV11??
017120  001456                          BEQ     6$              ;YES, SKIP TEST

017122  012737  000340  002242          MOV     #340,BDDAT      ;SET UP INITIAL OF 7
017130  013737  002146  002240          MOV     BPRIOR,GDDAT    ;GET GIVEN PRIORITY

017136                          BGNSEG                          ;****START OF SEGMENT****
017136  104004                          EMT     C$BSEG

017140  005037  002206          5$:     CLR     INTFLG          ;CLEAR INTERRUPT OCCURRANCE
017144                                  SETPRI  BDDAT           ;SET PRIORITY
017144  013700  002242                  MOV     BDDAT,R0
017150  104041                          EMT     C$SPRI

017152  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
017156  000100                          NOOP0!INTEN

017160  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
017164                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
017164  104010                          EMT     C$ESCAPE
017166  000070                          .WORD   L10046-.

017170  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
017174                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
017174  104010                          EMT     C$ESCAPE
017176  000060                          .WORD   L10046-.

017200  023737  002242  002240          CMP     BDDAT,GDDAT     ;SHOULD IT INTERRUPT
017206  002012                          BGE     1$              ;NO, BRANCH

017210  005737  002206                  TST     INTFLG          ;DID INTERRUPT OCCUR
017214  001004                          BNE     2$              ;YES, OK

017216                          3$:     ERRDF   204.,EM17,ERR7
017216  104462                          TRAP    T$ERCODE
017220  000314                          .WORD   204
017222  005323                          .WORD   EM17
017224  010364                          .WORD   ERR7

017226                          2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
017226  104010                          EMT     C$ESCAPE
017230  000014                          .WORD   10000$-.
017232  000405                          BR      4$
017234  005737  002206          1$:     TST     INTFLG          ;DID INTERRUPT OCCUR
017240  001772                          BEQ     2$              ;NO, OK
017242  000765                          BR      3$              ;YES, ERROR

017244                          ENDSEG                          ;****END OF SEGMENT****
017244                          10000$:
```

```
                017244  104005                          EMT     C$ESEG
679             017246  162737  000040  002242  4$:     SUB     #40,BDDAT       ;NEXT LEVEL
680             017254  100331                          BPL     5$
681
682             017256                          6$:
683             017256                          ENDTST                          ;****END OF TEST****
                017256                          L10046:
                017256  104001                          EMT     C$ETST
684
685                                             .SBTTL  **TEST 25** - GET STATUS FUNCTION
686
687             017260                          BGNTST                          ;****START OF TEST****
688
689
690             017260                          STARS
                                                ;;**********************************************************************
691                                             ;TEST GET STATUS FUNCTION.  THE GET STATUS FUNCTION WILL
692                                             ;WORK IF DRIVE IS LOADED AND READY OR NOT.  THE RLDA
693                                             ;IS LOADED WITH THE GET STATUS AND MARKER BITS (BITS 1,0)
694                                             ;AND THE FUNCTION IS ISSUED.  WE WAIT 200 MILLISECONDS
695                                             ;FOR CONTROLLER READY.  VERIFY THAT NO ERRORS OCCUR.
696             017260                          STARS
                                                ;;**********************************************************************
697
698
699             017260  012777  000013  162652          MOV     #GSBIT!MK!DRST,@RLDA    ;SET GET STATUS AND MARKER BIT
700             017266  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
701             017272  000004                          GSTAT                   ;GET STATUS
702             017274  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
703             017300                          2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                017300  104006                          EMT     C$CLP1
704
705             017302  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
706
707             017306                          ENDTST                          ;****END OF TEST****
                017306                          L10047:
                017306  104001                          EMT     C$ETST
708
709
710                                             .SBTTL  **TEST 26** - GET STATUS FUNCTION INTERRUPT
711
712             017310                          BGNTST                          ;****START OF TEST****
713
714                                             ;CHECK GET STATUS UNDER INTERRUPT
715
716             017310  005037  002206                  CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE
717             017314                          SETPRI  #PR100                  ;PSW TO LEVEL 0
                017314  012700  000000                  MOV     #PR100,R0
                017320  104041                          EMT     C$SPRI
718             017322  012777  000003  162610          MOV     #GSBIT!MK,@RLDA ;SET UP DA
719             017330  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
720             017334  000104                          GSTAT!INTEN             ;GET STATUS, INT ENABLE
721             017336  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
722             017342                          SETPRI  #PR107
                017342  012700  000340                  MOV     #PR107,R0
                017346  104041                          EMT     C$SPRI
723             017350  005737  002206                  TST     INTFLG          ;DID INTERRUPT OCCUR
```

```
                                        BNE     2$              ;YES-BRANCH
                                        ERRDF   28.,EM30,ERRO
                                        TRAP    T$ERCODE
                                        .WORD   28
                                        .WORD   EM30
                                        .WORD   ERRO
                                2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1
                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                        CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE
                                        SETPRI  #PR100          ;PSW TO LEVEL 0
                                        MOV     #PR100,R0
                                        EMT     C$SPRI
                                        MOV     #GSBIT!MK,@RLDA ;SET UP DA FOR GET STATUS CMD
                                        JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                        GSTAT                   ;GET STATUS - SHOULD NOT CAUSE AN INTERRUPT
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        SETPRI  #PR107
                                        MOV     #PR107,R0
                                        EMT     C$SPRI
                                        TST     INTFLG          ;DID INTERRUPT OCCUR (SHOULD NOT)
                                        BEQ     3$              ;NO - BRANCH (OK)
                                        ERRDF   281.,EM30A,ERRO
                                        TRAP    T$ERCODE
                                        .WORD   281
                                        .WORD   EM30A
                                        .WORD   ERRO
                                3$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1
                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                ENDTST                          ;****END OF TEST****
                                L10050:
                                        EMT     C$ETST


                                .SBTTL  **TEST 27** - GET STATUS FUNCTION GENERATES OPI W/O GS BIT

                                BGNTST                          ;****START OF TEST****

                                STARS
                                ;;**************************************************************
                                ;VERIFY THAT GET STATUS FUNCTION WILL NOT COMPLETE
                                ;WITHOUT SENDING OUT THE GET STATUS BIT IN THE RLDA.
                                ;WE SET MARKER BUT NO GET STATUS BIT IN THE RLDA AND
                                ;ISSUE A GET STATUS WE SHOULD RECIEVE AN OPI ERROR.
                                ;VERIFY THAT CONTROLLER READY SETS AND OPI SETS
                                STARS
                                ;;**************************************************************


                                        MOV     #MK,@RLDA       ;SET ONLY MARKER BIT!!
                                        JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                        GSTAT                   ;GET STATUS
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        BIT #74000,E.CS
                                        BEQ     1$
                                        MOV     #OPIERR,RESTMS
```

```
                                        JSR     R5,CHERR
                                1$:     CKLOOP
                                        EMT     C$CLP1
                                        BIT     #OPI,E.CS       ;IS OPI SET?
                                        BNE     2$              ;YES-BRANCH NO-CHECK TIMEOUT
                                        ERRDF   29.,EM33,ERRO
                                        TRAP    T$ERCODE
                                        .WORD   29
                                        .WORD   EM33
                                        .WORD   ERRO
                                2$:

                                ENDTST                          ;****END OF TEST****
                                L10051:
                                        EMT     C$ETST


                                .SBTTL  **TEST 28** - OPI UNDER INTERRUPT

                                BGNTST                          ;****START OF TEST****
                                STARS
                                ;;**************************************************************
                                ;FORCE AN OPI ERROR UNDER INTERRUPT TO VERIFY THAT
                                ;AN INTERRUPT WILL OCCUR FROM OPI.  THE OPI IS FORCED
                                ;USING A GET STATUS WITHOUT THE GET STATUS BIT SET
                                ;IN RLDA.
                                STARS
                                ;;**************************************************************


                                        SETPRI  #PRI00
                                        MOV     #PRI00,R0
                                        EMT     C$SPRI
                                        CLR     INTFLG
                                        MOV     #MK,@RLDA       ;SET ONLY MARKER BIT!!
                                        JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                        GSTAT!INTEN             ;GET STATUS
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        SETPRI  #PRI07
                                        MOV     #PRI07,R0
                                        EMT     C$SPRI
                                        TST     INTFLG          ;INTERRUPT OCCUR
                                        BNE     2$

                                        ERRDF   30.,EM11,ERRO
                                        TRAP    T$ERCODE
                                        .WORD   30
                                        .WORD   EM11
                                        .WORD   ERRO

                                2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1

                                        BIT     #74000,E.CS
                                        BEQ     1$
                                        MOV     #OPIERR,RESTMS
                                        JSR     R5,CHERR
```

```
802 017652                          1$:     CKLOOP
    017652  104006                          EMT     C$CLP1
803 017654  032737  002000  002166          BIT     #OPI,E.CS       ;IS OPI SET?
804 017662  001004                          BNE     3$              ;YES-BRANCH NO-CHECK TIMEOUT
805 017664                                  ERRDF   31.,EM33,ERR0
    017664  104462                          TRAP    T$ERCODE
    017666  000037                          .WORD   31
    017670  005511                          .WORD   EM33
    017672  010076                          .WORD   ERR0
806 017674                          3$:
807
808
809 017674                          ENDTST                          ;****END OF TEST****
    017674                          L10052:
    017674  104001                          EMT     C$ETST
810
811                                 .SBTTL  **TEST 29** - READ HEADER FUNCTION
812
813 017676                          BGNTST                          ;****START OF TEST****
814 017676                          STARS
                                    ;;**************************************************************
815                                 ;CHECK THAT READ HEADER WORKS, THAT WE CAN ISSUE
816                                 ;IT, GET READY BACK WITHOUT ANY ERRORS SETTING.
817 017676                          STARS
                                    ;;**************************************************************
818
819
820
821 017676  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
822 017702  000010                          RDHDR                   ;READ HEADER
823 017704  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
824                                                                 ;READY
825 017710                          2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
    017710  104006                          EMT     C$CLP1
826
827 017712  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
828
829 017716                          ENDTST                          ;****END OF TEST****
    017716                          L10053:
    017716  104001                          EMT     C$ETST
830
```

```
    1
```

```
                        .SBTTL  **TEST 30** - READ HEADER FUNCTION INTERRUPT
                        
017720                  BGNTST                          ;****START OF TEST****
                        
017720                  STARS
                        ;;*****************************************************************
                        ;CHECK THAT READ HEADER WILL GENERATE AN INTERRRUPT
                        ;UPON COMPLETION WITHOUT ANY ERRORS SETTING
017720                  STARS
                        ;;*****************************************************************
                        
                        
017720                          SETPRI  #PR100          ;PSW TO 0
017720  012700  000000          MOV     #PR100,R0
017724  104041                  EMT     C$SPRI
017726  005037  002206          CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE
017732  004537  013102          JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
017736  000110                  RDHDR!INTEN             ;READ HEADER, INTR. ENA
017740  004537  013712          JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
017744                          SETPRI  #PR107
017744  012700  000340          MOV     #PR107,R0
017750  104041                  EMT     C$SPRI
017752  005737  002206          TST     INTFLG          ;INTERRUPT HAPPEN
017756  001004                  BNE     2$              ;YES-CONTINUE
017760                          ERRDF   35.,EM37,ERR0
017760  104462                  TRAP    T$ERCODE
017762  000043                  .WORD   35
017764  005633                  .WORD   EM37
017766  010076                  .WORD   ERR0
017770                  2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
017770  104006                  EMT     C$CLP1
```

```
                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS

017776                          ENDTST                          ;****END OF TEST****
017776                          L10054:
017776  104001                          EMT     C$ETST


                                .SBTTL  **TEST 31** - REPEATED RD HDRS YIELD SAME CYL AND HD

020000                          BGNTST                          ;****START OF TEST****


020000                          STARS
                                ;;**************************************************************
                                ;CHECKT THAT READ HEADERS WILL RELIABLY READ THE SAME
                                ;CYLINDER AND HEAD SELECT.  WE WILL READ HEADERS VERIFYING
                                ;THAT WE ALWAYS READ THE SAME CYLINDER AND HEAD SELECT.
020000                          STARS
                                ;;**************************************************************


020000  012701  000144                  MOV     #100.,R1        ;SET UP TO DO 100 RD HDR'S
020004  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
020010  000010                          RDHDR                   ;READ HEADER
020012  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
020016                          99$:    ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
020016  104010                          EMT     C$ESCAPE
020020  000122                          .WORD   L10055-.

020022  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
020026                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
020026  104010                          EMT     C$ESCAPE
020030  000112                          .WORD   L10055-.

020032  013737  002174  002240          MOV     E.MP,GDDAT      ;READ FIRST HEADER (ASSUME GOOD)
020040  043737  002212  002240          BIC     SECMSK,GDDAT    ;MASK AWAY SECTOR BITS
020046                          BGNSEG                          ;****START OF SEGMENT****
020046  104004                          EMT     C$BSEG
020050                          2$:
020050  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
020054  000010                          RDHDR
020056  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
020062                          97$:    ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
020062  104010                          EMT     C$ESCAPE
020064  000054                          .WORD   10000$-.

020066  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
020072                                  ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
020072  104010                          EMT     C$ESCAPE
020074  000044                          .WORD   10000$-.

020076  013737  002174  002242          MOV     E.MP,BDDAT      ;READ HEADER
020104  043737  002212  002242          BIC     SECMSK,BDDAT    ;MASK AWAY SECTOR BITS
020112  023737  002240  002242          CMP     GDDAT,BDDAT     ;IS HEADER CORRECT
020120  001404                          BEQ     4$
```

```
                                        ERRDF   36.,EM41,ERR4
020122  104462                          TRAP    T$ERCODE
020124  000044                          .WORD   36
020126  005673                          .WORD   EM41
020130  010242                          .WORD   ERR4
                                                        ;CONSTANT CYL & HS
020132                          4$:     CKLOOP          ;CHECK IF /FL:LOE IS SET
020132  104006                          EMT     C$CLP1
                                        DEC     R1      ;PERFORM ALL READ HDR'S
020134  005301                          BNE     2$      ;IF NOT GO BACK AND DO ANOTHER
020136  001344                  ENDSEG                  ;****END OF SEGMENT****
020140                          10000$:
020140  104005                          EMT     C$ESEG
020142                          ENDTST                  ;****END OF TEST****
020142                          L10055:
020142  104001                          EMT     C$ETST


                                .SBTTL  **TEST 32** - CHECK OF HEADER CRC

                                BGNTST                  ;****START OF TEST****

020144                          STARS
                                ;;**********************************************************************
                                ;CHECK THAT WE CAN READ THE HDCRC AFTER A
                                ;READ HEADER AND THAT IT IS THE CORRECT CRC
                                ;FOR THE HEADER.
020144                          STARS
                                ;;**********************************************************************


020144  005037  020214                  CLR     3$
020150  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
020154  000010                          RDHDR                   ;READ HEADER
020156  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
020162                          ESCAPE  TST                     ;IF /FL:LOE SET LOOP, ELSE EXIT TST
020162  104010                          EMT     C$ESCAPE
020164  000114                          .WORD   L10056-.

020166  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
020172                          ESCAPE  TST                     ;IF /FL:LOE SET LOOP, ELSE EXIT TST
020172  104010                          EMT     C$ESCAPE
020174  000104                          .WORD   L10056-.

020176  013737  002174  020212          MOV     E.MP,2$         ;READ HEADER WORD
020204  004537  013460                  JSR     R5,SIMBCC       ;GO CALCULATE CRC
020210  000020                          16.                     ;16 BITS
020212  000000                  2$:     .WORD   0               ;HEADER GOES HERE
020214  000000                  3$:     .WORD   0               ;START WITH 0 CRC
020216  013737  002222  020242          MOV     CALBCC,5$
020224  013737  002176  020240          MOV     E.MP1,4$        ;GET SECOND HALF
020232  004537  013460                  JSR     R5,SIMBCC
020236  000020                          16.
020240  000000                  4$:     .WORD   0
020242  000000                  5$:     .WORD   0
020244  013737  002222  002240          MOV     CALBCC,GDDAT    ;STORE CALCULATED CRC AS GOOD
```

```
     020252  013737  002200  002242          MOV     E.MP2,BDDAT     ;THIRD READ OF DB GETS CRC
     020260  023737  002240  002242          CMP     GDDAT,BDDAT     ;IS CRC CORRECT?
     020266  001404                          BEQ     6$              ;IF SO CONTINUE

     020270                                  ERRDF   37.,EM42,ERR4
     020270  104462                          TRAP    T$ERCODE
     020272  000045                          .WORD   37
     020274  005764                          .WORD   EM42
     020276  010242                          .WORD   ERR4
     020300                          6$:

     020300                          ENDTST                          ;****END OF TEST****
     020300                          L10056:
     020300  104001                          EMT     C$ETST


                                     .SBTTL  **TEST 33** - CHECK CONSECUTIVE HEADERS

     020302                          BGNTST                          ;****START OF TEST****


     020302                          STARS
                                     ;;*****************************************************************
                                     ;CHECK THAT THE HEADERS ARE CONSECUTIVE.  WE WILL DO
                                     ;40 (FORTY) READ HEADERS AND STORE EACH.  AFTER WE HAVE
                                     ;READ THE FORTIETH HEADER WE WILL VERIFY THAT
                                     ;THEY CAME IN SEQUENTIAL, THAT 0 FOLLOWS 39,
                                     ;THAT THERE WERE NO ERRORS.
     020302                          STARS
                                     ;;*****************************************************************


     020302  005037  002244                  CLR     FIRST           ;CLEAR FIRST READ DONE FLAG
     020306  012703  003150                  MOV     #HDRBUF,R3      ;STORE HEADERS
     020312  012701  000050                  MOV     #40.,R1         ;FOURTY HEADERS
     020316  012737  000210  002154          MOV     #RDHDR!CRDY,B.CS
     020324  053737  002152  002154          BIS     DRIVE,B.CS
     020332  013777  002154  161574          MOV     B.CS,@RLCS
     020340  042777  000200  161566  2$:     BIC     #200,@RLCS
     020346  032777  000200  161560  1$:     BIT     #200,@RLCS      ;DONE?
     020354  001774                          BEQ     1$
     020356  017723  161552                  MOV     @RLCS,(R3)+
     020362  017723  161554                  MOV     @RLMP,(R3)+
     020366  017723  161550                  MOV     @RLMP,(R3)+
     020372  017723  161544                  MOV     @RLMP,(R3)+
     020376  005301                          DEC     R1              ;HAVE WE READ FOURTY HEADERS
     020400  001357                          BNE     2$              ;GO BACK UNTIL FOURTY DONE
     020402  012703  003150                  MOV     #HDRBUF,R3      ;GET LIST OF HEADERS
     020406  012701  000050                  MOV     #40.,R1         ;CHECK FOURTY OF THEM
     020412  011337  002166                  MOV     (R3),E.CS
     020416  005737  002166                  TST     E.CS
     020422  100016                          BPL     99$
     020424  012737  004227  013064          MOV     #RHDMES,RESTMS
     020432  005723                          TST     (R3)+
     020434  012337  002174                  MOV     (R3)+,E.MP
     020440  012337  002176                  MOV     (R3)+,E.MP1
     020444  012337  002200                  MOV     (R3)+,E.MP2
```

```
                                                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                                JMP     7$
                                        99$:    TST     (R3)+
                                                MOV     (R3),BDDAT      ;GET HEADER
                                                TST     FIRST           ;IS THIS FIRST READ?
                                                BNE     4$              ;NO, BRANCH
                                                MOV     #1,FIRST        ;SET FIRST READ DONE FLAG
                                        3$:     MOV     BDDAT,GDDAT     ;SET UP NEXT READ EXPECTED
                                                BR      6$              ;GO SEE IF TEST IS DONE
                                        4$:     INC     GDDAT           ;INCREMENT EXP'D HEADER
                                                CMP     BDDAT,GDDAT     ;IS NEW HEADER SEQUENTIAL?
                                                BEQ     3$              ;YES THEN BRANCH
                                                BIT     SECMSK,BDDAT    ;IS NEW HEADER ZERO?
                                                BNE     5$              ;NO, THEN ERROR GO REPORT IT
                                                MOV     GDDAT,TEMP2     ;YES, CHECK IF LAST HEADER WAS
                                                BIC     CYLMSK,TEMP2    ;MAX ADDRESS, IF SO BRANCH
                                                CMP     MXSEC1,TEMP2    ;STORE NEW DATA AS OLD
                                                BEQ     3$              ;AND PERFORM NEW RD HDR
                                                BIC     SECMSK,GDDAT    ;EXPECTING ZERO SECTOR

                                        5$:

                                                CLR     FIRST           ;ERROR WILL MAKE US MISS
                                                                        ;NEXT SECTOR SEQUENTIALLY
                                                                        ;START OVER; CLEAR FIRST FLAG
                                                ERRDF   38.,EM43,ERR2
                                                TRAP    T$ERCODE
                                                .WORD   38
                                                .WORD   EM43
                                                .WORD   ERR2
                                        6$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                                EMT     C$CLP1

                                                ADD     #6,R3
                                                DEC     R1              ;HAVE WE DONE THIS ENOUGH
                                                BNE     99$             ;NO, GO BACK DO IT AGAIN
                                        7$:
                                        ENDTST                          ;****END OF TEST****
                                        L10057:
                                                EMT     C$ETST


                                        .SBTTL  **TEST 34** - SEEK FUNCTION

                                        BGNTST                          ;****START OF TEST****
                                        STARS
                                        ;;*****************************************************************
                                        ;CHECK THE SEEK FUNCTION RETURNS CONTROLLER READY
                                        ;WITH NO ERRORS.  WE ISSUE A ONE TRACK IN WORD SEEK.
                                        ;WE DO NOT CHECK THE RESULT FOR POSITION
                                        STARS
                                        ;;*****************************************************************


                                                MOV     #BIT7!MK!SIGN,@RLDA  ;SET UP DA-DIFF=1,MARKER,TOWARDS
                                                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                                SEEK                    ;SEEK
```

```
020634  004537  013712                    JSR     R5,WTCRDY         ;WAIT FOR CONTROLLER READY HIGH
020640                                    WAITMS  #2.
020640  012700  000002                    MOV     #2.,R0
020644  104026                            EMT     C$WTM
020646                            2$:     CKLOOP                    ;CHECK IF /FL:LOE IS SET
020646  104006                            EMT     C$CLP1


020650  004537  012602                    JSR     R5,CHERR          ;CHECK CONTROLLER FOR ERRORS

020654                            ENDTST                            ;****END OF TEST****
020654                            L10060:
020654  104001                            EMT     C$ETST


                                  .SBTTL  **TEST 35** - CHECK DRIVE READY ON SEEK

020656                            BGNTST                            ;****START OF TEST****


020656                            STARS
                                  ;;*************************************************************
                                  ;CHECK THE SEEK FUNCTION RETURNS DRIVE READY WITH
                                  ;NO ERRORS.  WE ISSUE A ONE TRACK INWARD SEEK.  WE DO
                                  ;NOT CHECK THE RESULT FOR POSITION
020656                            STARS
                                  ;;*************************************************************



020656  012777  000201  161254            MOV     #BIT7!MK,@RLDA    ;SET DA, MARKER, DIFF=1.
020664  004537  013102                    JSR     R5,LDFUNC         ;ISSUE FUNCTION OF FOLLOWING WORD
020670  000006                            SEEK                      ;SEEK
020672  004537  013712                    JSR     R5,WTCRDY         ;WAIT FOR CONTROLLER READY HIGH
020676                                    CKLOOP                    ;CHECK IF /FL:LOE IS SET
020676  104006                            EMT     C$CLP1

020700  004537  012602                    JSR     R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
020704                                    CKLOOP                    ;CHECK IF /FL:LOE IS SET
020704  104006                            EMT     C$CLP1

020706  004537  013646                    JSR     R5,WTDRDY         ;WAIT FOR DRIVE READY
020712                                    CKLOOP                    ;CHECK IF /FL:LOE IS SET
020712  104006                            EMT     C$CLP1

020714  004537  012602                    JSR     R5,CHERR          ;CHECK CONTROLLER FOR ERRORS

020720                            ENDTST                            ;****END OF TEST****
020720                            L10061:
020720  104001                            EMT     C$ETST


                                  .SBTTL  **TEST 36** - SEEK FUNCTION INTERRUPT

020722                            BGNTST                            ;****START OF TEST****


```

```
    229 020722                          STARS
                                        ;;*****************************************************************
    230                                 ;CHECK THAT CONTROLLER READY RESETTING WHEN THE SEEK IS
    231                                 ;INITIATED CAUSES AN INTERRUPT BUT DRIVE READY WILL
    232                                 ;NOT.  WE ALSO MONITOR FOR ANY ERROR BITS SETTING.
    233 020722                          STARS
                                        ;;*****************************************************************
    234
    235
    236
    237
    238 020722  005037  002206                  CLR     INTFLG
    239 020726                          SETPRI  #PR100          ;SET PSW TO 0
        020726  012700  000000          MOV     #PR100,R0
        020732  104041                  EMT     C$SPRI
    240 020734  012777  000205  161176  MOV     #BIT7!MK!SIGN,@RLDA  ;SET UP RLDA
    241 020742  004537  013102          JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
    242 020746  000106                  SEEK!INTEN              ;SEEK AND INTR. ENA.
    243 020750  004537  013712          JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
    244 020754  000240                  NOP                     ;
    245 020756  005737  002206  1$:     TST     INTFLG          ;DID INTERRUPT OCCUR
    246 020762  001004                  BNE     2$              ;YES, GO CHECK DRDY
    247 020764                          ERRDF   40.,EM47,ERR0
        020764  104462                  TRAP    T$ERCODE
        020766  000050                  .WORD   40
        020770  006242                  .WORD   EM47
        020772  010076                  .WORD   ERR0
    248 020774                  2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
        020774  104006                  EMT     C$CLP1
    249
    250
    251 020776  004537  012602          JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
    252 021002                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
        021002  104006                  EMT     C$CLP1
    253
    254 021004  005037  002206          CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE
    255
    256
    257 021010  004537  013646          JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
    258 021014                  5$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
        021014  104006                  EMT     C$CLP1
    259
    260 021016                          SETPRI  #PR107
        021016  012700  000340          MOV     #PR107,R0
        021022  104041                  EMT     C$SPRI
    261 021024  005737  002206          TST     INTFLG          ;DID DRIVE READY CAUSE INTERRUPT
    262 021030  001404                  BEQ     6$              ;NO, CONTINUE
    263
    264 021032                          ERRDF   42.,EM52,ERR0
        021032  104462                  TRAP    T$ERCODE
        021034  000052                  .WORD   42
        021036  006273                  .WORD   EM52
        021040  010076                  .WORD   ERR0
    265 021042                  6$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
        021042  104006                  EMT     C$CLP1
    266
    267 021044                  ENDTST                          ;****END OF TEST****
```

```
                021044
                021044  104001                  L10062:
                                                        EMT     C$ETST
    268
    269
    270                                         .SBTTL  **TEST 37** - TEST DIFFERENCE WORD TRANSMISSION
    271
    272 021046                                  BGNTST                          ;****START OF TEST****
    273
    274
    275
    276
    277 021046                                  STARS
                                                ;;**************************************************************
    278                                         ;VERIFY THAT THE DIFFERENCE WORD LOADS AND IS
    279                                         ;TRANSMITTED CORRECTLY.  WE WILL ISSUE SEEKS WITH THE
    280                                         ;DIFFERENCE WORD CONTAINING ALL OF THE BIT PATTERNS FLOATING 1,
    281                                         ;GROWING 1, GROWING 0 AND SHITING 0.  THE SEEK WILL
    282                                         ;START FROM TRACK 0 EACH TIME AND WILL RETURN THERE
    283                                         ;EACH, THUS BOTH DIRECTIONS FOR PATTERNS WILL BE CHECKED.
    284                                         ;READ HEADERS ARE USED TO VERIFY THE SEEK CORRECTNESS.
    285                                         ;ERRORS ARE MONITORED AND REPORTED.
    286 021046                                  STARS
                                                ;;**************************************************************
    287
    288
    289 021046  012703  002502                          MOV     #SKLST,R3       ;GET LIST OF DIFFERENCE WORDS
    290 021052                                  BGNSEG                          ;****START OF SEGMENT****
                021052  104004                          EMT     C$BSEG
    291 021054                                  1$:
    292 021054  004537  013102                          JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
    293 021060  000010                                  RDHDR                   ;READ HEADER
    294 021062  004537  013712                          JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
    295 021066                                  98$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
                021066  104006                          EMT     C$CLP1
    296
    297 021070  004537  012602                          JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
    298 021074                                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
                021074  104006                          EMT     C$CLP1
    299
    300 021076  013737  002174  002242                  MOV     E.MP,BDDAT      ;READ HEADER
    301 021104  043737  002212  002242                  BIC     SECMSK,BDDAT    ;CLEAR OUT SECTOR
    302 021112  001462                                  BEQ     99$             ;IF ON TRACK ZERO, H.S. ZERO, OK
    303
    304                                         ;NOT ON TRACK ZERO CALCULATE DIFFERENCE WORD AND PUT IT BACK
    305                                         ;ON ZERO.
    306
    307 021114  042737  000100  002242                  BIC     #RHHS,BDDAT     ;CLEAR OUT HEAD SELECT
    308 021122  013777  002242  161010                  MOV     BDDAT,@RLDA     ;PUT CYLINDER AS DIFFERENCE WORD
    309 021130  052777  000001  161002                  BIS     #MK,@RLDA       ;SET MARKER BIT
    310 021136  004537  013102                          JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
    311 021142  000006                                  SEEK                    ;SEEK
    312 021144  004537  013712                          JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
    313 021150                                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
                021150  104006                          EMT     C$CLP1
    314
    315 021152  004537  012602                          JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
    316 021156                                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
```

```
     021156  104006                          EMT     C$CLP1
317
318 021160  004537  013646                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
319 021164                          89$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021164  104006                          EMT     C$CLP1
320
321 021166  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
322 021172                                   CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021172  104006                          EMT     C$CLP1
323
324 021174  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
325 021200  000010                           RDHDR                   ;READ HEADER
326 021202  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
327 021206                          96$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021206  104006                          EMT     C$CLP1
328
329 021210  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
330 021214                                   CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021214  104006                          EMT     C$CLP1
331
332 021216  005037  002240                  CLR     GDDAT           ;CLEAR EXPECTED
333 021222  013737  002242  002254          MOV     BDDAT,DWORD     ;SAVE DIFFERENCE WORD
334 021230  013737  002174  002242          MOV     E.MP,BDDAT      ;READ HEADER
335 021236  043737  002212  002242          BIC     SECMSK,BDDAT    ;MASK OUT SECTOR BITS
336 021244  001404                          BEQ     5$              ;BRANCH IF ON ZERO TRACK
337
338 021246                                   ERRDF   43.,EM54,ERR3
     021246  104462                          TRAP    T$ERCODE
     021250  000053                          .WORD   43
     021252  006343                          .WORD   EM54
     021254  010170                          .WORD   ERR3
339 021256                          5$:      CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021256  104006                          EMT     C$CLP1
340
341 021260  011377  160654          99$:     MOV     (R3),@RLDA      ;GET DIFFERENCE WORD
342 021264  052777  000005  160646          BIS     #SIGN!MK,@RLDA  ;SET SIGN (TOWARDS SPINDLE) AND MARKER
343 021272  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
344 021276  000006                           SEEK                    ;SEEK
345 021300  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
346 021304                                   CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021304  104006                          EMT     C$CLP1
347
348 021306  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
349 021312                                   CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021312  104006                          EMT     C$CLP1
350
351 021314  004537  013646                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
352 021320                          87$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021320  104006                          EMT     C$CLP1
353
354 021322  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
355 021326                                   CKLOOP                  ;CHECK IF /FL:LOE IS SET
     021326  104006                          EMT     C$CLP1
356
357 021330  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
358 021334  000010                           RDHDR                   ;READ HEADER
359
```

```
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        CKLOOP                  ;CHECK IF /FL:LOE IS SET
021342  104006                          EMT     C$CLP1

                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                        ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
021350  104010                          EMT     C$ESCAPE
021352  000106                          .WORD   10000$-.

                                        MOV     (R3),GDDAT      ;GET EXPECTED CYLINDER
                                8$:     MOV     (R3),DWORD      ;SET UP DIFFERENCE FOR SEEK
                                        MOV     E.MP,BDDAT      ;READ HEADER FROM RLMP
                                        BIC     SECMSK,BDDAT    ;CLEAR OUT SECTOR BITS
                                        CMP     GDDAT,BDDAT     ;DID SEEK GO TO THE RIGHT
                                        BEQ     9$              ;TRACK, IF SO, GO GET NEXT

                                        ERRDF   44.,EM54,ERR3
021410  104462                          TRAP    T$ERCODE
021412  000054                          .WORD   44
021414  006343                          .WORD   EM54
021416  010170                          .WORD   ERR3
                                9$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
021420  104006                          EMT     C$CLP1

                                        TST     (R3)+           ;BUMP PATTERN
                                        CMP     T.DRIVE,#1
                                        BNE     2$
                                        CMP     R3,#SKEND
                                        BEQ     10$
                                        JMP     1$

                                2$:     CMP     R3,#SKEEND
                                        BEQ     10$
                                        JMP     1$

                                10$:

                                ENDSEG                          ;****END OF SEGMENT****
                                10000$:
                                        EMT     C$ESEG
                                ENDTST                          ;****END OF TEST****
                                L10063:
                                        EMT     C$ETST


                                .SBTTL  **TEST 38** - VERIFY HEAD SELECT 0 VIA RD HDR

                                BGNTST                          ;****START OF TEST****

                                ;

                                STARS
                                ;;*****************************************************************
                                ;CHECK THAT WE CAN SELECT HEAD SELECT ZERO.  ISSUE
                                ;SEEK TO HEAD SELECT 0 AND VERIFY WITH READ HEADER.
                                STARS
                                ;;*****************************************************************
```

```
021464  012777  000001  160446  99$:    MOV     #MK,@RLDA           ;SET MARKER IN RLDA
021472  005037  002240                  CLR     GDDAT               ;SET EXPECTED
                                                                    ;LOAD HS=0 INTO RLDA
021476                          2$:
021476  004537  013102                  JSR     R5,LDFUNC           ;ISSUE FUNCTION OF FOLLOWING WORD
021502  000006                          SEEK                        ;SEEK
021504  004537  013712                  JSR     R5,WTCRDY           ;WAIT FOR CONTROLLER READY HIGH
021510                                  CKLOOP                      ;CHECK IF /FL:LOE IS SET
021510  104006                          EMT     C$CLP1

021512  004537  012602                  JSR     R5,CHERR            ;CHECK CONTROLLER FOR ERRORS
021516                                  CKLOOP                      ;CHECK IF /FL:LOE IS SET
021516  104006                          EMT     C$CLP1

021520  004537  013646                  JSR     R5,WTDRDY           ;WAIT FOR DRIVE READY
021524                          89$:    CKLOOP                      ;CHECK IF /FL:LOE IS SET
021524  104006                          EMT     C$CLP1

021526  004537  012602                  JSR     R5,CHERR            ;CHECK CONTROLLER FOR ERRORS
021532                                  CKLOOP                      ;CHECK IF /FL:LOE IS SET
021532  104006                          EMT     C$CLP1

021534  004537  013102                  JSR     R5,LDFUNC           ;ISSUE FUNCTION OF FOLLOWING WORD
021540  000010                          RDHDR                       ;READ HEADER
021542  004537  013712                  JSR     R5,WTCRDY           ;WAIT FOR CONTROLLER READY HIGH
021546                          96$:    CKLOOP                      ;CHECK IF /FL:LOE IS SET
021546  104006                          EMT     C$CLP1

021550  004537  012602                  JSR     R5,CHERR            ;CHECK CONTROLLER FOR ERRORS
021554                                  ESCAPE  TST                 ;IF /FL:LOE SET LOOP, ELSE EXIT TST
021554  104010                          EMT     C$ESCAPE
021556  000036                          .WORD   L10064-.

021560  013737  002174  002242          MOV     E.MP,BDDAT          ;READ HEADER FOR HEAD SELECT
021566  042737  177677  002242          BIC     #177677,BDDAT       ;MASK ONLY HEAD SELECT
021574  023737  002240  002242          CMP     GDDAT,BDDAT         ;COMPARE HEAD SELECTS
021602  001404                          BEQ     5$                  ;IF EQUAL CONTINUE

021604                                  ERRDF   45.,EM55,ERR4
021604  104462                          TRAP    T$ERCODE
021606  000055                          .WORD   45
021610  006402                          .WORD   EM55
021612  010242                          .WORD   ERR4
021614                          5$:

021614                          ENDTST                              ;****END OF TEST****
021614                          L10064:
021614  104001                          EMT     C$ETST



                                .SBTTL  **TEST 39** - VERIFY HEAD SELECT 1 VIA RD HDR

021616                          BGNTST                              ;****START OF TEST****


021616                          STARS
```

```
                                ;;*****************************************************************
                                ;CHECK THAT WE CAN SELECT HEAD SELECT ONE.  ISSUE
                                ;SEEK TO HEAD SELECT 1 AND VERIFY WITH READ HEADER.
021616                          STARS
                                ;;*****************************************************************

021616  012777  000001  160314 99$:    MOV     #MK,@RLDA       ;SET MARKER IN RLDA
021624  052777  000020  160306         BIS     #DAHS,@RLDA     ;LOAD HS=1 INTO RLDA
021632  004537  013102          2$:    JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
021636  000006                         SEEK                    ;SEEK
021640  004537  013712                 JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
021644                                 CKLOOP                  ;CHECK IF /FL:LOE IS SET
021644  104006                         EMT     C$CLP1

021646  004537  012602                 JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
021652                                 CKLOOP                  ;CHECK IF /FL:LOE IS SET
021652  104006                         EMT     C$CLP1

021654  004537  013646                 JSR     R5,WTDRDY       ;WAIT FOR DRIVE CLEAR
021660                         89$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
021660  104006                         EMT     C$CLP1

021662  004537  012602                 JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
021666                                 CKLOOP                  ;CHECK IF /FL:LOE IS SET
021666  104006                         EMT     C$CLP1

021670  004537  013102                 JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
021674  000010                         RDHDR                   ;READ HEADER
021676  004537  013712                 JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
021702                                 CKLOOP                  ;CHECK IF /FL:LOE IS SET
021702  104006                         EMT     C$CLP1

021704  004537  012602                 JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
021710                         ESCAPE  TST                     ;IF /FL:LOE SET LOOP, ELSE EXIT TST
021710  104010                         EMT     C$ESCAPE
021712  000044                         .WORD   L10065-.

021714  013737  002174  002242         MOV     E.MP,BDDAT      ;READ HEADER
021722  042737  177677  002242         BIC     #177677,BDDAT   ;MASK FOR H.S.
021730  012737  000100  002240         MOV     #RHHS,GDDAT     ;SET EXPECTED
021736  023737  002240  002242         CMP     GDDAT,BDDAT     ;CORRECT HEAD
021744  001404                         BEQ     5$              ;YES, CONTINUE

021746                                 ERRDF   46.,EM55,ERR4
021746  104462                         TRAP    T$ERCODE
021750  000056                         .WORD   46
021752  006402                         .WORD   EM55
021754  010242                         .WORD   ERR4
021756                          5$:

021756                          ENDTST                          ;****END OF TEST****
021756                          L10065:
021756  104001                         EMT     C$ETST


                                .SBTTL  **TEST 40** - VERIFY HEAD SELECT 0 VIA GET STATUS
```

```
                                BGNTST                          ;****START OF TEST****

                                STARS
                                ;;*****************************************************************
                                ;CHECK THAT WE CAN READ BACK HEAD SELECT 0 WITH
                                ;A GET STATUS FUNCTION.  SELECT H.S. 0 WITH A SEEK
                                ;VERIFY WITH GET STATUS
                                STARS
                                ;;*****************************************************************

022760  012777  000001  160152          MOV     #MK,@RLDA       ;SET MARKER IN RLDA
                                                                  ;LOAD HS=0 INTO RLDA
022766  005037  002240          2$:     CLR     GDDAT           ;SET UP EXP'D
022772  004537  013102          3$:     JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
022776  000006                          SEEK                    ;SEEK
023000  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
023004                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
        023004  104006                  EMT     C$CLP1

023006  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
023012                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
        023012  104006                  EMT     C$CLP1

023014  004537  013646                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
023020                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
        023020  104006                  EMT     C$CLP1

023022  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
023026                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
        023026  104006                  EMT     C$CLP1

023030  012777  000003  160102          MOV     #GSBIT!MK,@RLDA ;SET UP FOR GET STATUS IN DA
023036  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
023042  000004                          GSTAT                   ;GET STATUS
023044  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
023050                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
        023050  104006                  EMT     C$CLP1

023052  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
023056                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
        023056  104010                  EMT     C$ESCAPE
        023060  000036                  .WORD   L10066-.

023062  013737  002174  002242          MOV     E.MP,BDDAT      ;READ STATUS FOR HEAD SELECT BIT
023070  042737  177677  002242          BIC     #177677,BDDAT   ;LEAVE ONLY H.S. BIT
023076  023737  002240  002242          CMP     GDDAT,BDDAT     ;IS HEAD SELECT CORRECT?
023104  001404                          BEQ     6$              ;YES, CONTINUE

023106                                  ERRDF   47.,EM56,ERR4
        023106  104462                  TRAP    T$ERCODE
        023110  000057                  .WORD   47
        023112  006435                  .WORD   EM56
        023114  010242                  .WORD   ERR4
023116                          6$:

023116                          ENDTST                          ;****END OF TEST****
```

```
                                L10066:
 022116
 022116  104001                         EMT     C$ETST
533
534
535                             .SBTTL  **TEST 41** - VERIFY HEAD SELECT 1 VIA GET STATUS
536
537 022120                      BGNTST                          ;****START OF TEST****
538
539 022120                      STARS
                                ;;**************************************************************************
540                             ;CHECK THAT WE CAN READ BACK HEAD SELECT 1 WITH A GET
541                             ;STATUS FUNCTION.  SELECT H.S. 1 WITH A SEEK AND VERIFIY WITH
542                             ;GET STATUS
543 022120                      STARS
                                ;;**************************************************************************
544
545
546 022120  012777  000001  160012          MOV     #MK,@RLDA       ;SET MARKER IN RLDA
547 022126  052777  000020  160004          BIS     #DAHS,@RLDA     ;LOAD HS=1 INTO RLDA
548 022134  012737  000100  002240  2$:     MOV     #STHS,GDDAT     ;SET UP EXP'D
549 022142  004537  013102          3$:     JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
550 022146  000006                          SEEK                    ;SEEK
551 022150  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
552 022154                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 022154  104006                             EMT     C$CLP1
553
554 022156  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
555 022162                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 022162  104006                             EMT     C$CLP1
556
557 022164  004537  013646                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
558 022170                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 022170  104006                             EMT     C$CLP1
559
560 022172  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
561 022176                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 022176  104006                             EMT     C$CLP1
562
563 022200  012777  000003  157732          MOV     #GSBIT!MK,@RLDA ;SET UP FOR GET STATUS IN DA
564 022206  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
565 022212  000004                          GSTAT                   ;GET STATUS
566 022214  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
567 022220                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
 022220  104010                             EMT     C$ESCAPE
 022222  000046                             .WORD   L10067-.
568
569 022224  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
570 022230                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
 022230  104010                             EMT     C$ESCAPE
 022232  000036                             .WORD   L10067-.
571
572 022234  013737  002174  002242          MOV     E.MP,BDDAT      ;READ STATUS FOR HEAD SELECT BIT
573 022242  042737  177677  002242          BIC     #177677,BDDAT   ;LEAVE ONLY H.S. BIT
574 022250  023737  002240  002242          CMP     GDDAT,BDDAT     ;IS HEAD SELECT CORRECT?
575 022256  001404                          BEQ     6$              ;YES, CONTINUE
576
577 022260                                  ERRDF   48.,EM56,ERR4
```

```
        022260  104462                          TRAP    T$ERCODE
        022262  000060                          .WORD   48
        022264  006435                          .WORD   EM56
        022266  010242                          .WORD   ERR4
    578 022270                          6$:
    579
    580 022270                          ENDTST                          ;****END OF TEST****
        022270                          L10067:
        022270  104001                          EMT     C$ETST
    581
    582
    583                                 .SBTTL  **TEST 42** - TEST TIME AT WHICH DIF WD GETS TRANSMITTED
    584
    585 022272                          BGNTST                          ;****START OF TEST****
    586
    587
    588 022272                          STARS
                                        ;;*****************************************************************
    589                                 ;VERIFY THAT THE DIFFERENCE WORD ON A SEEK IS
    590                                 ;TRANSMITTED PRIOR TO CONTROLLER READY SETTING.  THIS
    591                                 ;IS DONE BY SETTING A KNOWN DIFFERENCE WORD IN
    592                                 ;THE RLDA ISSUING A A SEEK, WAITING FOR CONTROLLER READY
    593                                 ;(BUT NOT DRIVE READY), WRITING A DIFFERENT RLDA AND WAITING
    594                                 ;FOR DRIVE READY.  THE RESULTANT POSITION SHOULD BE THAT
    595                                 ;OF THE FIRST RLDA ONLY.
    596 022272                          STARS
                                        ;;*****************************************************************
    597
    598
    599 022272  004537  013102                      JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
    600 022276  000010                              RDHDR                   ;READ HEADER
    601 022300  004537  013712                      JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
    602 022304                          99$:        CKLOOP                  ;CHECK IF /FL:LOE IS SET
        022304  104006                              EMT     C$CLP1
    603
    604 022306  004537  012602                      JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
    605 022312                                      CKLOOP                  ;CHECK IF /FL:LOE IS SET
        022312  104006                              EMT     C$CLP1
    606
    607 022314  013737  002174  002240              MOV     E.MP,GDDAT      ;READ HEADER
    608 022322  043737  002212  002240              BIC     SECMSK,GDDAT    ;CLEAR SECTOR BITS
    609 022330  012777  000001  157602              MOV     #MK,@RLDA       ;SET MARKER IN RLDA
    610 022336  032737  000100  002240              BIT     #RHHS,GDDAT     ;TEST H.S.
    611 022344  001403                              BEQ     2$              ;IF ZERO, CONTINUE
    612 022346  052777  000020  157564              BIS     #DAHS,@RLDA     ;ONE, SET SO WE WILL REMAIN THERE
    613 022354  013737  002240  002232  2$:         MOV     GDDAT,TMP0      ;STORE HEADER
    614 022362  042737  000100  002232              BIC     #RHHS,TMP0      ;CLEAR H.S. FROM STORED WORD
    615 022370  023727  002264  000001              CMP     T.DRIVE,#1
    616 022376  001034                              BNE     12$
    617 022400  023737  002232  002560              CMP     TMP0,HALMAX
    618 022406  101007                              BHI     3$
    619 022410  052777  000004  157522              BIS     #SIGN,@RLDA
    620 022416  063737  002556  002240              ADD     QUAMAX,GDDAT
    621 022424  000403                              BR      4$
    622 022426  163737  002556  002240  3$:         SUB     QUAMAX,GDDAT
    623 022434  053777  002556  157476  4$:         BIS     QUAMAX,@RLDA
    624 022442  012737  000001  002234              MOV     #MK,TMP1
```

```
022450  032777  000020  157462          BIT     #DAHS,@RLDA
022456  001037                          BNE     5$
022460  052737  000020  002234          BIS     #DAHS,TMP1
022466  000433                          BR      5$
022470  023737  002232  002610  12$:    CMP     TMP0,HMAX
022476  101007                          BHI     13$
022500  052777  000004  157432          BIS     #SIGN,@RLDA
022506  063737  002606  002240          ADD     QMAX,GDDAT
022514  000403                          BR      14$
022516  163737  002606  002240  13$:    SUB     QMAX,GDDAT
022524  053777  002606  157406  14$:    BIS     QMAX,@RLDA
022532  012737  000001  002234          MOV     #MK,TMP1
022540  032777  000020  157372          BIT     #DAHS,@RLDA
022546  001003                          BNE     5$
022550  052737  000020  002234          BIS     #DAHS,TMP1
022556  004537  013102          5$:     JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
022562  000006                          SEEK                    ;SEEK
022564  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
022570                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
022570  104006                          EMT     C$CLP1


022572  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
022576                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
022576  104006                          EMT     C$CLP1

022600  013777  002234  157332          MOV     TMP1,@RLDA      ;SEND IN NEW DIFFERENCE WORD
022606  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
022612                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
022612  104006                          EMT     C$CLP1

022614  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
022620                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
022620  104006                          EMT     C$CLP1

022622  004537  013646                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
022626                          8$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
022626  104006                          EMT     C$CLP1


022630  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
022634                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
022634  104006                          EMT     C$CLP1

022636  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
022642  000010                          RDHDR                   ;READ HEADER
022644  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
022650                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
022650  104006                          EMT     C$CLP1

022652  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
022656                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
022656  104010                          EMT     C$ESCAPE
022660  000036                          .WORD   L10070-.

022662  013737  002174  002242          MOV     E.MP,BDDAT      ;READ HEADER
022670  043737  002212  002242          BIC     SECMSK,BDDAT    ;CLEAR SECTOR ADDRESS
```

```
022676  023737  002240  002242          CMP     GDDAT,BDDAT         ;IS HEADER CORRECT?
022704  001404                          BEQ     10$                 ;IF SO BRANCH

022706                                  ERRDF   50.,EM57,ERR4
022706  104462                          TRAP    T$ERCODE
022710  000062                          .WORD   50
022712  006474                          .WORD   EM57
022714  010242                          .WORD   ERR4
022716                          10$:

022716                          ENDTST                              ;****END OF TEST****
022716                          L10070:
022716  104001                          EMT     C$ETST



                                .SBTTL  **TEST 43** - EXTENSIVE CHECK OF HEADER CRC

022720                          BGNTST                              ;****START OF TEST****
022720                          STARS
                                ;;*************************************************************
                                ;MORE EXTENSIVE CHECK OF HEADER CRC.  WE WILL SEEK
                                ;AND READ HEADERS VERIFYING HDR CRC ACROSS THE
                                ;PLATTER USING THE GROWING 0, GROWING 1, SHIFTING 0 AND
                                ;GROWING 0 PATTERNS FOR TRACK ADDRESSES.
022720                          STARS
                                ;;*************************************************************


022720  012703  002502                  MOV     #SKLST,R3           ;GET LIST OF DIFFERENCE WORDS
022724                          BGNSEG                              ;****START OF SEGMENT****
022724  104004                          EMT     C$BSEG
022726                          1$:
022726  004537  013102                  JSR     R5,LDFUNC           ;ISSUE FUNCTION OF FOLLOWING WORD
022732  000010                          RDHDR                       ;READ HEADER
022734  004537  013712                  JSR     R5,WTCRDY           ;WAIT FOR CONTROLLER READY HIGH
022740                          98$:    CKLOOP                      ;CHECK IF /FL:LOE IS SET
022740  104006                          EMT     C$CLP1

022742  004537  012602                  JSR     R5,CHERR            ;CHECK CONTROLLER FOR ERRORS
022746                                  CKLOOP                      ;CHECK IF /FL:  E IS SET
022746  104006                          EMT     C$CLP1

022750  013737  002174  002242          MOV     E.MP,BDDAT          ;READ HEADER
022756  043737  002212  002242          BIC     SECMSK,BDDAT        ;CLEAR OUT SECTOR
022764  001461                          BEQ     5$                  ;IF ON TRACK ZERO, H.S. ZERO, OK

                                ;NOT ON TRACK ZERO CALCULATE DIFFERENCE WORD AND PUT IT BACK
                                ;ON ZERO.

022766  042737  000100  002242          BIC     #RHHS,BDDAT         ;CLEAR OUT HEAD SELECT
022774  013777  002242  157136          MOV     BDDAT,@RLDA         ;PUT CYLINDER AS DIFFERENCE WORD
023002  052777  000001  157130          BIS     #MK,@RLDA           ;SET MARKER BIT
023010  004537  013102                  JSR     R5,LDFUNC           ;ISSUE FUNCTION OF FOLLOWING WORD
023014  000006                          SEEK                        ;SEEK
023016  004537  013712                  JSR     R5,WTCRDY           ;WAIT FOR CONTROLLER READY HIGH
023022                                  CKLOOP                      ;CHECK IF /FL:LOE IS SET
023022  104006                          EMT     C$CLP1
```

```
                                                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
023030  104006                                  EMT     C$CLP1

                                                JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
                                        89$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
023036  104006                                  EMT     C$CLP1

                                                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
023044  104006                                  EMT     C$CLP1


                                                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                                RDHDR                   ;READ HEADER
                                                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        96$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
023060  104006                                  EMT     C$CLP1

                                                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
023066  104006                                  EMT     C$CLP1

023070  005037  002240                          CLR     GDDAT           ;CLEAR EXPECTED
023074  013737  002242  002254                  MOV     BDDAT,DWORD     ;SAVE DIFFERENCE WORD
023102  013737  002174  002242                  MOV     E.MP,BDDAT      ;READ HEADER
023110  043737  002212  002242                  BIC     SECMSK,BDDAT    ;MASK OUT SECTOR BITS
023116  001404                                  BEQ     5$              ;BRANCH IF ON ZERO TRACK

023120                                          ERRDF   51.,EM54,ERR3
023120  104462                                  TRAP    T$ERCODE
023122  000063                                  .WORD   51
023124  006343                                  .WORD   EM54
023126  010170                                  .WORD   ERR3
023130                                  5$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
023130  104006                                  EMT     C$CLP1

023132  011377  157002                          MOV     (R3),@RLDA      ;GET DIFFERENCE WORD
023136  052777  000005  156774                  BIS     #SIGN!MK,@RLDA  ;SET SIGN (TOWARDS SPINDLE) AND MARKER
023144  004537  013102                          JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
023150  000006                                  SEEK                    ;SEEK
023152  004537  013712                          JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
023156                                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
023156  104006                                  EMT     C$CLP1

023160  004537  012602                          JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
023164                                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
023164  104006                                  EMT     C$CLP1

023166  004537  013646                          JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
023172                                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
023172  104006                                  EMT     C$CLP1


023174  004537  012602                          JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
023200                                          CKLOOP                  ;CHECK IF /FL:LOE IS SET
```

```
023200  104006                              EMT     C$CLP1

023202  004537  013102                      JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
023206  000010                              RDHDR                   ;READ HEADER
023210  004537  013712                      JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
023214                                      CKLOOP                  ;CHECK IF /FL:LOE IS SET
023214  104006                              EMT     C$CLP1


023216  004537  012602                      JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
023222                                      CKLOOP                  ;CHECK IF /FL:LOE IS SET
023222  104006                              EMT     C$CLP1

023224  011337  002240                      MOV     (R3),GDDAT      ;GET EXPECTED CYLINDER
023230  011337  002254              8$:     MOV     (R3),DWORD      ;SET UP DIFFERENCE FOR SEEK
023234  013737  002174  002242              MOV     E.MP,BDDAT      ;READ HEADER FROM RLMP
023242  043737  002212  002242              BIC     SECMSK,BDDAT    ;CLEAR OUT SECTOR BITS
023250  023737  002240  002242              CMP     GDDAT,BDDAT     ;DID SEEK GO TO THE RIGHT
023256  001404                              BEQ     9$              ;TRACK, IF SO, GO GET NEXT

023260                                      ERRDF   52.,EM54,ERR3
023260  104462                              TRAP    T$ERCODE
023262  000064                              .WORD   52
023264  006343                              .WORD   EM54
023266  010170                              .WORD   ERR3
023270                              9$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
023270  104006                              EMT     C$CLP1

023272  013737  002174  023306              MOV     E.MP,10$        ;GET HEADER WORD
023300  004537  013460                      JSR     R5,SIMBCC       ;GO CALCULATE HEADER CRC
023304  000020                              16.                     ;16 BITS
023306  000000                      10$:    .WORD   0               ;HEADER GOES HERE
023310  000000                              .WORD   0               ;START WITH ZERO CRC
023312  013737  002222  023336              MOV     CALBCC,20$
023320  013737  002176  023334              MOV     E.MP1,21$
023326  004537  013460                      JSR     R5,SIMBCC
023332  000020                              16.
023334  000000                      21$:    .WORD   0
023336  000000                      20$:    .WORD   0
023340  013737  002222  002240              MOV     CALBCC,GDDAT    ;MOVE CALCULATED CRC TO GDDAT
023346  013737  002200  002242              MOV     E.MP2,BDDAT     ;GET HEADER CRC FROM RLMP
023354  023737  002240  002242              CMP     GDDAT,BDDAT     ;IS CRC CORRECT?
023362  001404                              BEQ     11$             ;IF SO CONTINUE

023364                                      ERRDF   53.,EM42,ERR4
023364  104462                              TRAP    T$ERCODE
023366  000065                              .WORD   53
023370  005764                              .WORD   EM42
023372  010242                              .WORD   ERR4
023374                              11$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
023374  104006                              EMT     C$CLP1


023376  005723                              TST     (R3)+           ;BUMP PATTERN
023400  023727  002264  000001              CMP     T.DRIVE,#1
023406  001005                              BNE     2$
023410  020327  002602                      CMP     R3,#SKEND
```

```
                                        BEQ     12$
                                        JMP     1$
                                2$:     CMP     R3,#SKEEND
                                        BEQ     12$
                                        JMP     1$
                                12$:

                                ENDSEG                          ;****END OF SEGMENT****
                                10000$:
                                        EMT     C$ESEG
                                ENDTST                          ;****END OF TEST****
                                L10071:
                                        EMT     C$ETST


                                .SBTTL  **TEST 44** - VERIFY GET STATUS WHILE DRDY IS LOW

                                BGNTST                          ;****START OF TEST****

                                STARS
                                ;;*****************************************************************
                                ;VERIFY THAT WE CAN ISSUE GET STATUS AND RECIEVE
                                ;THE STATUS WORD WHILE THE DRIVE IS IN NOTION SEEKING
                                STARS
                                ;;*****************************************************************


                                1$:
                                        JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                        RDHDR                   ;READ HEADER
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1

                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                        CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1

                                        MOV     E.MP,BDDAT      ;READ HEADER
                                        BIC     SECMSK,BDDAT    ;CLEAR OUT SECTOR
                                        BEQ     5$              ;IF ON TRACK ZERO, H.S. ZERO, OK

                                ;NOT ON TRACK ZERO CALCULATE DIFFERENCE WORD AND PUT IT BACK
                                ;ON ZERO.

                                        BIC     #RHHS,BDDAT     ;CLEAR OUT HEAD SELECT
                                        MOV     BDDAT,@RLDA     ;PUT CYLINDER AS DIFFERENCE WORD
                                        BIS     #MK,@RLDA       ;SET MARKER BIT
                                        JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                        SEEK                    ;SEEK
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1

                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                        CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1
```

```
    023544  004537  013646                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
    023550                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
    023550  104006                          EMT     C$CLP1

    023552  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
    023556                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
    023556  104006                          EMT     C$CLP1


    023560  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
    023564  000010                          RDHDR                   ;READ HEADER
    023566  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
    023572                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
    023572  104006                          EMT     C$CLP1

    023574  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
    023600                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
    023600  104006                          EMT     C$CLP1

    023602  005037  002240                  CLR     GDDAT           ;CLEAR EXPECTED
    023606  013737  002242  002254          MOV     BDDAT,DWORD     ;SAVE DIFFERENCE WORD
    023614  013737  002174  002242          MOV     E.MP,BDDAT      ;READ HEADER
    023622  043737  002212  002242          BIC     SECMSK,BDDAT    ;MASK OUT SECTOR BITS
    023630  001404                          BEQ     5$              ;BRANCH IF ON ZERO TRACK

    023632                                  ERRDF   54.,EM54,ERR3
    023632  104462                          TRAP    T$ERCODE
    023634  000066                          .WORD   54
    023636  006343                          .WORD   EM54
    023640  010170                          .WORD   ERR3
    023642                          5$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
    023642  104006                          EMT     C$CLP1

    023644  012777  077601  156266          MOV     #77601,@RLDA    ;GET DIFFERENCE WORD
    023652  052777  000005  156260          BIS     #SIGN!MK,@RLDA  ;SET SIGN (TOWARDS SPINDLE) AND MARKER
    023660  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
    023664  000006                          SEEK                    ;SEEK
    023666  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
    023672                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
    023672  104006                          EMT     C$CLP1


    023674  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
    023700                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
    023700  104006                          EMT     C$CLP1
    023702  012777  000003  156230          MOV     #MK!GSBIT,@RLDA
    023710  004537  013102                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
    023714  000004                          GSTAT
    023716  004537  013712                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
    023722                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
    023722  104006                          EMT     C$CLP1
    023724  004537  012602                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS

    023730                          ENDTST                          ;****END OF TEST****
    023730                          L10072:
    023730  104001                          EMT     C$ETST
```

```
                                        BGNMOD  HRDPRM

                                                BGNHRD
023732  000030                                  .WORD L10073-L$HARD/2

023734                                          GPRML   CNTMSG,CNT,1,YES
023734  005130                                  .WORD   T$CODE
023736  024030                                  .WORD   CNTMSG
023740  000001                                  .WORD   1
023742                                          GPRMA   CSRMSG,CSR,0,160000,177776,YES
023742  000031                                  .WORD   T$CODE
023744  024014                                  .WORD   CSRMSG
023746  160000                                  .WORD   T$LOLIM
023750  177776                                  .WORD   T$HILIM
023752                                          GPRMA   VECMSG,VECT,0,0,776,YES
023752  001031                                  .WORD   T$CODE
023754  024046                                  .WORD   VECMSG
023756  000000                                  .WORD   T$LOLIM
023760  000776                                  .WORD   T$HILIM
023762                                          GPRMD   BRMSG,PRIOR,0,340,0,7,YES
023762  002032                                  .WORD   T$CODE
023764  024035                                  .WORD   BRMSG
023766  000340                                  .WORD   340
023770  000000                                  .WORD   T$LOLIM
023772  000007                                  .WORD   T$HILIM
023774                                          GPRML   DRTYPE,TYPDR,1,YES
023774  003130                                  .WORD   T$CODE
023776  024055                                  .WORD   DRTYPE
024000  000001                                  .WORD   1
024002                                          GPRMD   DRMSG,DRBT,0,03400,0,7,YES
024002  004032                                  .WORD   T$CODE
024004  024077                                  .WORD   DRMSG
024006  003400                                  .WORD   03400
024010  000000                                  .WORD   T$LOLIM
024012  000007                                  .WORD   T$HILIM

024014                                          ENDHRD
                                                .EVEN
024014                          L10073:

024014  102  125  123  CSRMSG: .ASCIZ  /BUS ADDRESS/
024017  040  101  104
024022  104  122  105
024025  123  123  000
024030  122  114  061  CNTMSG: .ASCIZ  /RL11/
024033  061  000
024035  102  122  040  BRMSG:  .ASCIZ  /BR LEVEL/
024040  114  105  126
024043  105  114  000
024046  126  105  103  VECMSG: .ASCIZ  /VECTOR/
024051  124  117  122
024054  000
024055  104  122  111  DRTYPE: .ASCIZ  /DRIVE TYPE = RL01/
024060  126  105  040
024063  124  131  120
024066  105  040  075
```

```
                024071  040     122     114
                024074  060     061     000
    915         024077  104     122     111     DRMSG:  .ASCIZ  /DRIVE/
                024102  126     105     000
    916                                         .EVEN
    917
    918         024106                          ENDMOD
    919
    920
    921
    922         024106                          BGNMOD  SFTPRM
    923
    924         024106                          BGNSFT
                024106  000014                  .WORD L10074-L$SOFT/2
    925         024110                          GPRML   DMSG,DLT,1,YES
                024110  000130                  .WORD   T$CODE
                024112  024140                  .WORD   DMSG
                024114  000001                  .WORD   1
    926         024116                          XFERF   1$
                024116  006044                  .WORD   T$CODE
    927         024120                          GPRMD   EMSG,ELT,0,177777,0,177777,YES
                024120  001032                  .WORD   T$CODE
                024122  024175                  .WORD   EMSG
                024124  177777                  .WORD   177777
                024126  000000                  .WORD   T$LOLIM
                024130  177777                  .WORD   T$HILIM
    928         024132                  1$:     GPRML   SMSG,SIZE,1,YES
                024132  002130                  .WORD   T$CODE
                024134  024164                  .WORD   SMSG
                024136  000001                  .WORD   1
    929         024140                          ENDSFT
                                                .EVEN
                024140                  L10074:
    930
    934
    935         024140      104     122     117     DMSG:   .ASCIZ  /DROP ON ERROR LIMIT/
    936         024164      101     125     124     SMSG:   .ASCIZ  /AUTOSIZE/
    937         024175      105     122     122     EMSG:   .ASCIZ  /ERROR LIMIT/
    938
    942
    943                                         .EVEN
    944
    945         024212                          ENDMOD
    946                 024514                  .=24514         ;THIS FIXES THE 'APT' MAILBOX ADDRESS
    947         024514                  LASTAD
                                                .EVEN
                024514                  L$LAST::
    948
```

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

```
     35 055310  000000                    .WORD   0               ;SPACE FOR USER POOL POINTER
     36 055312  000000                    .WORD   0               ;SIZE
     37 055314  000000                    .WORD   0               ;CHECKSUM (NOT CURRENTLY USED)
     38 055316  000000                    .WORD   0               ;SIZE OF H.W. PTAB. ALLOCATION
     39         055322                    END.SUPV=.+2
     40         000200                    .END 200
```

```
ABOFLA  025040 G    BLD.HW  032202    CURR.S  024522 G    C$SVEC= 000037    EF09  = 000011 G
ABOPAS  024756 G    BLOCK   047614    CURR.T  024524 G    C$TPRI= 000013    EF10  = 000012 G
ABO.FM  027320      BPRIOR  002146    CYLMSK  002246      C$UNBU= 000031    EF11  = 000013 G
ADDCOD  012512 G    BRMSG   024035    C$AAD   037062      C$WTM = 000026    EF12  = 000014 G
AFREG   004561      BVEC    002150    C$AAE   037074      C$WTU = 000027    EF13  = 000015 G
AFSI    024546 G    B$AAB   033604    C$AAK   040072      DAHS  = 000020    EF14  = 000016 G
AFTER   013412      B$AAF   033516    C$AAL   040236      DATEST  014600    EF15  = 000017 G
ALLOC   045460      B.BA    002156    C$ABRT= 000021      DCKMES  003743    EF16  = 000020 G
APT.ER  026450      B.CS    002154    C$ADR = 000020      DECMSG  044004    ELT   = 000002
ARLBA   004516      B.DA    002160    C$AU  = 000054      DEMES   003711    EMSG    024175
ARLCS   004511      B.MP    002162    C$BRK = 000022      DERFLG  002164    EMT.TR  025044 G
ARLDA   004524      CALBCC  002222    C$BSEG= 000004      DERR  = 040000    EM1     004655
ARLMP   004532      CALLPC= 000022    C$BSUB= 000002      DEV.CO  024526 G    EM101   007640
ASSEMB= 000011      CALLPS= 000024    C$BUFF= 000030      DIAGMC= 000000    EM102   007705
A$AAV   031316      CALLSP= 000026    C$CEFG= 000046      DIAG.T  025046 G    EM11    005126
A$AAW   031332      CALLTC= 000030    C$CLEA= 000012      DLT   = 000000    EM13    005167
A$AAX   031344      CAL.CL  052202    C$CLP1= 000006      DLTMES  003750    EM14    005221
A$AAY   031352      CAL.TI  052240 G  C$CVEC= 000036      DMSG    024140    EM15    005247
A$AAZ   031366      CHERR   012602    C$DCLN= 000044      DPDVD   054456 G    EM16    005275
A$ABA   031376      CHKLUP  033620    C$DODU= 000053      DPMUL   054344 G    EM17    005323
BATEST  014476      CHKSTR  046022    C$DRPT= 000024      DRBT  = 000010    EM2     004702
BA16  = 000020      CHKTTY  044110    C$DU  = 000055      DRDY  = 000001    EM3     004727
BA17  = 000040      CHK.MA  031760    C$EDIT= 000002      DRIVE   002152    EM30    005356
BCCFBK  002220      CHK.PC  037110    C$ERDF= 000002      DRMSG   024077    EM30A   005415
BCSR    002144      CHK.SW  026150    C$ERHR= 000003      DROP    011436    EM32    005455
BDDAT   002242      CHRCNT  045342    C$ERSF= 000001      DRPCOD  012506 G    EM33    005511
BEFORE  013360      CH.FLA  031466    C$ERSO= 000004      DRST  = 000010    EM34    005556
BEGPAT  002272      CH.PAS  031504    C$ESCA= 000010      DRTIM   004627    EM37    005633
BEREG   004540      CKERLT  012516    C$ESEG= 000005      DRTYPE  024055    EM4     004754
BGN.SU= 024514      CLEAR.  033102    C$ESUB= 000003      DSPCOD  011444 G    EM41    005673
BINMSG  043770      CLKACC  024754 G  C$ETST= 000001      DS0   = 000000    EM42    005764
BIT0  = 000001 G    CLKBFR  052204    C$EXIT= 000032      DS1   = 000400    EM43    006022
BIT00 = 000001 G    CLKCNT  024752 G  C$GMAN= 000043      DS2   = 001000    EM44    006121
BIT01 = 000002 G    CLKJUM  052610 G  C$GPHR= 000042      DS3   = 001400    EM45    006154
BIT02 = 000004 G    CLKRES  053612 G  C$GPRI= 000040      DUNIT.  024762 G    EM46    006207
BIT03 = 000010 G    CLKSER  053746 G  C$GTIM= 000052      DVC.FT  040042    EM47    006242
BIT04 = 000020 G    CLKSON  025012 G  C$INIT= 000011      DWORD   002254    EM5     005001
BIT05 = 000040 G    CLK.SE  031562    C$INLP= 000020      D$AAG   040746    EM52    006273
BIT06 = 000100 G    CLNCOD  012440 G  C$KWOF= 000035      D$AAH   040764    EM54    006343
BIT07 = 000200 G    CLR.MA  032036    C$KWON= 000034      D$AAI   043532    EM55    006402
BIT08 = 000400 G    CNT   = 000012    C$LOOP= 000100      D$AAJ   043536    EM56    006435
BIT09 = 001000 G    CNTMSG  024030    C$MANI= 000051      D$AAK   043554    EM57    006474
BIT1  = 000002 G    CNVT    050260    C$MSG = 000023      D$AAL   043572    EM6     005052
BIT10 = 002000 G    COMMAN  024564 G  C$PNTB= 000014      D$AAM   043602    EM61    006575
BIT11 = 004000 G    COMMTA  050074    C$PNTF= 000017      EF.CON= 000036 G    EM62    006656
BIT12 = 010000 G    COMP    003762    C$PNTS= 000016      EF.NEW= 000035 G    EM63    006741
BIT13 = 020000 G    CONT    012012    C$PNTX= 000015      EF.PWR= 000034 G    EM64    007022
BIT14 = 040000 G    CONTCL  053672 G  C$POIN= 000040      EF.RES= 000037 G    EM65    007105
BIT15 = 100000 G    CONTIN  011660    C$QIO = 000377      EF.STA= 000040 G    EM66    007166
BIT2  = 000004 G    CRDY  = 000200    C$RDBU= 000007      EF01  = 000001 G    EM67    007251
BIT3  = 000010 G    CRLF    044172    C$REFG= 000050      EF02  = 000002 G    EM7     005100
BIT4  = 000020 G    CRTIM   004602    C$REQT= 000045      EF03  = 000003 G    EM70    007306
BIT5  = 000040 G    CSEND   002744    C$RESE= 000033      EF04  = 000004 G    EM71    007343
BIT6  = 000100 G    CSPAT   002646    C$REVI= 000002      EF05  = 000005 G    EM72    007400
BIT7  = 000200 G    CSR   = 000000    C$RPT = 000025      EF06  = 000006 G    EM73    007433
BIT8  = 000400 G    CSRMSG  024014    C$SEFG= 000047      EF07  = 000007 G    EM74    007466
BIT9  = 001000 G    CSTEST  014356    C$SPRI= 000041      EF08  = 000010 G    EM75    007520
```

```
EM76    007552          FRMT4   011100          G$YES = 000010          LOAD.F  031502          L10001  010124
EM77    007605          FRMT5   011136          HALMAX  002560          LOGMSG  044012          L10002  010166
END     012374          FRMT6   011207          HCORED  031256          LPBFR   024622 G        L10003  010240
ENDPAT  002500          FRMT99  011133          HCOREQ  031166          LPCNTR  024620 G        L10004  010306
END.OF  033070          F$AU    = 000015        HCORET  025002 G        LPT.AD  031144          L10005  010320
END.SU= 055322          F$BGN   = 000040        HCRCME  003730          LPT.RE  031140          L10006  010362
ENVIRO  024566 G        F$CLEA= 000007          HC.ADR  024552 G        LSI.RE  031134          L10007  010420
EOP.CH  053770 G        F$DU    = 000016        HC.DEF  024544 G        LUP     052106          L10010  011434
EOP.FM  027334          F$END   = 000041        HC.DIA  024542 G        LUP.AD  037112          L10011  011444
EOP.IN  031500          F$HARD= 000004          HDRBUF  003150          L$APT   002036 G        L10012  012436
ERCOUN  002750          F$HW    = 000013        HDRLST  013322          L$AU    012512 G        L10013  012504
ERPOIN  002746          F$INIT= 000006          HERTZ.  031126          L$AUT   002074 G        L10014  012510
ERR     = 100000        F$JMP   = 000050        HMAX    002610          L$CCP   002106 G        L10015  012514
ERRFOR  040314          F$MOD   = 000000        HNFMES  003736          L$CLEA  012440 G        L10016  013644
ERRHAN  037114          F$MSG   = 000011        HOLDSP= 000020          L$CO    002032 G        L10017  014062
ERRVEC  002216          F$PWR   = 000017        HPTCOD  011416 G        L$DEPO  002011 G        L10020  014156
ERR.HR  040052          F$RPT   = 000012        HRDPRM  023732 G        L$DESC  002102 G        L10021  014252
ERR.NU  024516 G        F$SEG   = 000003        HW.ADR  024550 G        L$DEVP  002064 G        L10022  014346
ERR.SF  040056          F$SOFT= 000005          H$AAB   050606          L$DISP  011446 G        L10023  014466
ERR0    010076 G        F$SRV   = 000010        ININIT  024772 G        L$DR    002112 G        L10024  014570
ERR1    010114 G        F$SUB   = 000002        INITCO  011576 G        L$DRCT  002070 G        L10025  014656
ERR1FO  040400          F$SW    = 000014        INITIA  044020          L$DRS   002072 G        L10026  015002
ERR2    010126 G        F$TEST= 000001          INIT.M  032104          L$DRST  002112 G        L10027  015126
ERR3    010170 G        GARBAG  045344          INIT.R  024606 G        L$DTP   002040 G        L10030  015232
ERR4    010242 G        GDDAT   002240          INPUTA  044746          L$DU    012506 G        L10031  015332
ERR5    010310 G        GETCHR  044050          INTEN = 000100          L$DUT   002076 G        L10032  015422
ERR6    010322 G        GETCMN  047434          INTFLG  002206          L$DVTY  002114 G        L10033  015522
ERR7    010364 G        GETPAR  041126          INTFOR  040244          L$EF    002056 G        L10034  015632
ESC.PC  037106          GETSWI  046430          INTSRV  013640          L$EFLG  002034 G        L10035  015704
EV.COU  024520 G        GET.TW  046200          INVAL.  031052          L$EXP1  002042 G        L10036  015742
E.BA    002170          GLBDAT  002126 G        INVINT  040102          L$EXP2  002044 G        L10037  016066
E.CS    002166          GLBEQA  002126 G        INV.SW  026104          L$EXP3  002046 G        L10040  016226
E.DA    002172          GLBERR  010076 G        IN.SUF  033054          L$HARD  023734 G        L10041  016366
E.MP    002174          GLBSUB  012516 G        I$AU    = 000041        L$HPCP  002016 G        L10042  016572
E.MP1   002176          GLBTXT  003650 G        I$CLN   = 000041        L$HPTP  002022 G        L10043  016622
E.MP2   002200          GODRVR= 000202          I$DU    = 000041        L$HW    011420 G        L10044  017026
FILL    044640          GSBIT   = 000002        I$HRD   = 000041        L$ICP   002104 G        L10045  017112
FILL.C  000204 G        GSTAT   = 000004        I$INIT= 000041          L$INIT  011576 G        L10046  017256
FIRST   002244          GSTINT  004452          I$MOD   = 000041        L$LADP  002026 G        L10047  017306
FIX     013346          GSTMES  004413          I$MSG   = 000041        L$LAST  024514 G        L10050  017460
FLAGS   024560 G        G$EXCP= 000400          I$PWR   = 000041        L$MREV  002050 G        L10051  017546
FLAGS1  024562 G        G$HILI= 000002          I$RPT   = 000041        L$NAME  002000 G        L10052  017674
FLAGTA  050012          G$LOLI= 000001          I$SEG   = 000041        L$REPP  002066 G        L10053  017716
FLAG.I  031546          G$NO    = 000000        I$SFT   = 000041        L$REV   002010 G        L10054  017776
FLA.SE  047760          G$OFFS= 000400          I$SRV   = 000041        L$SOFT  024110 G        L10055  020142
FLG.MA  031506          G$OFSI= 000376          I$SUB   = 000041        L$SPC   002062 G        L10056  020300
FNDFNC  013320          G$PRMA= 000001          I$TST   = 000041        L$SPCP  002020 G        L10057  020616
FORM.T  040410          G$PRMD= 000002          J$JMP   = 000167        L$SPTP  002024 G        L10060  020654
FREE    045716          G$PRML= 000000          KBPTR   024624 G        L$STA   002030 G        L10061  020720
FRMT1   010752          G$RADA= 000140          KBUF    024626 G        L$SW    011436 G        L10062  021044
FRMT11  011244          G$RADB= 000000          LDCSR   002210          L$TIML  002014 G        L10063  021462
FRMT12  011305          G$RADD= 000040          LDFUNC  013102          L$TIMU  002054 G        L10064  021614
FRMT13  011364          G$RADF= 000200          LF      003760          L$TIM1  002052 G        L10065  021756
FRMT2   011012          G$RADL= 000120          LINE.F  025042 G        L$TSTI  002100 G        L10066  022116
FRMT2A  011031          G$RADO= 000020          LINE1   010422          L$UNIT  002012 G        L10067  022270
FRMT2B  011044          G$RADT= 000100          LINE2   010456          L.CLK.  031112          L10070  022716
FRMT3   011073          G$XFER= 000004          LINE3   010700          L10000  010112          L10071  023436
```

ASSEMBLY ROUTINES MACRO V03.01 9-FEB-79 18:53:41 PAGE 156-3
SYMBOL TABLE SEQ 0108

```
L10072  023730          O$GNSW= 000001          SFTPRM  024106 G        TYPEC   044336          T20     016370 G
L10073  024014          O$POIN= 000001          SHIFT   054776 G        TYPEPC  040232          T21     016574 G
L10074  024140          PARSES  047506          SIGN  = 000004        TYPFLA  047654          T22     016624 G
MAJ.IN  024576 G        PAR.LA  043474          SIMBCC  013460        TYPLIN  044234          T23     017030 G
MAJ.LO  052206          PASS.C  024530 G        SIZE  = 000004        TYPNUM  043616          T24     017114 G
MAJ.US  024600 G        PFLG    002202          SKEEND  002644        TYPSTR  044254          T25     017260 G
MAN.TI  001244          PRINTC  045320          SKEND   002602        TYP.ER  040062          T26     017310 G
MAP16   054714 G        PRINTF  050626          SKLST   002502        TY.UNI  033074          T27     017462 G
MASK.B  033616          PRIOR = 000004          SMSG    024164        T$ARGC= 000001          T28     017550 G
MASK.W  033614          PRI00 = 000000 G        SPEC.U  031406        T$CODE= 002130          T29     017676 G
MAXCYL  002256          PRI01 = 000040 G        SPTCOD  011434 G      T$ERCO= 000062          T3      014160 G
MAXSEC  002252          PRI02 = 000100 G        SPV.SE  000400        T$ERRN= 000066          T30     017720 G
MDHEDR  002000 G        PRI03 = 000140 G        START   011676        T$EXCP= 000000          T31     020000 G
MEM.SI  031154          PRI04 = 000200 G        STARTC  053666 G      T$FLAG= 000040          T32     020144 G
MERLMT  011440          PRI05 = 000240 G        START1  011640        T$HILI= 177777          T33     020302 G
MIN.IN  024572 G        PRI06 = 000300 G        STHS  = 000100        T$LOLI= 000000          T34     020620 G
MIN.US  024574 G        PRI07 = 000340 G        STRCHR  044700        T$LSYM= 010000          T35     020656 G
MK    = 000001          PRNTST  045210          STRT.T  031464        T$NEST= 177777          T36     020722 G
MODR    054256 G        PRO.CM  031460          ST.SET  026316        T$NSK0= 000000          T37     021046 G
MSCRLF  003755          PTAB.S  025000 G        SUNIT.  031470        T$NSK1= 000005          T38     021464 G
MSG.AD  024540 G        PUTCHR  044024          SUPERV  027352        T$SAVL= 177777          T39     021616 G
MSG.TY  024514 G        PWRFLG  002126          SUPFLA  024760 G      T$SEGL= 177777          T4      014254 G
MUL     054212 G        PWR.FA  055150 G        SUPV.T  025132 G      T$SEK0= 010000          T40     021760 G
MXSEC1  002250          PWR.FL  024604 G        SUP.PR  026070        T$SUBN= 000000          T41     022120 G
NEWPRI  053736 G        PWR.MS  055276          SVCGBL= 000000        T$TAGL= 177777          T42     022272 G
NEXTAR  050176          PWR.SA  055272          SVCHAN  034006        T$TAGN= 010075          T43     022720 G
NODRY   003666          PWR.UP  055274          SVCINS= 000000        T$TEMP= 000000          T44     023440 G
NOOP0 = 000000          P.CLK.  031120          SVCSUB= 177777        T$TEST= 000054          T5      014350 G
NOOP7 = 000016          QMAX    002606          SVCTAG= 000000        T$TSTM= 177777          T6      014470 G
NOPINT  004074          QUAMAX  002556          SVCTST= 177777        T$TSTS= 000001          T7      014572 G
NOPMES  004043          RDHDR = 000010          SVHD    002260        T$$AU = 010015          T8      014660 G
NOPWR   011620          READ  = 000014          SWCHAN  031300        T$$CLE= 010013          T9      015004 G
NORES   003650          READ.P  052210 G        SWITCH  050152        T$$DU = 010014          UNITST  002132
NO.CLK  031102          REGBAC  054700 G        SW.ADR  024554 G      T$$HAR= 010073          UNIT.D  024532 G
NO.FLA  047772          REGSAV  054664 G        SW.PTA  031264        T$$HW = 010010          UNI.MA  031410
NO.LPT  045310          REQN.P  024570 G        SYS.FT  040032        T$$INI= 010012          USER.P  024774 G
NO.PTA  031306          REQN.T  031462          S$LSYM= 010000        T$$MSG= 010007          USER.T  024776 G
NR    = 000000          REST    011736          TEMP2   002224        T$$SEG= 010000          UUT     002130
NUMBIN  040434          RESTMS  013064          TEMP3   002226        T$$SOF= 010074          VALID.  025234
NUM.LA  040602          RE.SET  026252          TEMP4   002230        T$$SRV= 010016          VAL.LA  026054
NUM.NO  024556 G        RHDINT  004267          TERMI   052176        T$$SW = 010011          VAL.SW  031520
NUM.UN  025164          RHDMES  004227          TERMLI  050000        T$$TES= 010072          VECMSG  024046
NUNITS  033572          RHHS  = 000100          TERMTA  043762        T.CNTL  002266          VECT  = 000002
NXM   = 020000          RLBA    002136          TEST.M  031420        T.DRIV  002264          WCKINT  004166
NXMMES  003716          RLCS    002134          TIMFLG  024750 G      T.SIZE  011442          WCKMES  004126
NXT     011670          RLDA    002140          TIM.CO  024602 G      T1      013770 G        WHY     002262
NXTFOR  050252          RLMP    002142          TIM.OP  040406        T10     015130 G        WIDTH   041002
OCTMSG  043776          RL2     002604          TMPFNC  002270        T11     015234 G        WRCHK = 000002
OKHDR   013332          RSTACK  054140 G        TMP0    002232        T12     015334 G        WRITE = 000012
OPI   = 002000          SAVEDO= 026450          TMP1    002234        T13     015424 G        WTCRDY  013712
OPIERR  003770          SEARCH  046146          TMP2    002236        T14     015524 G        WTDRDY  013646
OPIMES  003723          SECMSK  002212          TOO.MA  043742        T15     015634 G        XEQDIA  054024 G
O$APTS= 000000          SEEK  = 000006          TRPFLG  002204        T16     015706 G        XEQSUB  054012 G
O$AU  = 000001          SEGSTA  025014 G        TRPHAN  013632        T17     015744 G        XEQ.CL  033534
O$BGNR= 000000          SEKINT  004361          TST.AB  033730        T18     016070 G        XEQ.CM  031044
O$BGNS= 000001          SEKMES  004330          TST.TO  026132        T19     016230 G        XEQ.IN  033216
O$DU  = 000001          SET.MA  031672          TYPDR = 000006        T2      014064 G        XEQ.LA  027306
```

```
ASSEMBLY ROUTINES        MACRO V03.01 9-FEB-79 18:53:41 PAGE 156-4
SYMBOL TABLE                                                    SEQ 0109

XEQ.OP  033310          XTIME   052676 G        XXX     011720          X$TRUE= 000020          $SAV3   055056 G
XEQ.PR  026510          XTIMEN  053522         X$ALWA= 000000          $BREG   031560          $SAV4   055074 G
XEQ.TE  033354          XTIMST  052720         X$FALS= 000040          $ENDAD  053776 G        $SAV5   055114 G
XPOLY   002214          XXDP.D  031064         X$OFFS= 000400          $SAV2   055042 G

. ABS.  055320      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  19080 WORDS  ( 75 PAGES)
DYNAMIC MEMORY AVAILABLE FOR  70 PAGES
CZRLGA.BIN,CZRLGA=#SVCRT/M,CZRLGA,DOCTOR
```