RL11/RLV11,RL01

CONTROLLER TEST PART 1

MD-11-DZRLA-A

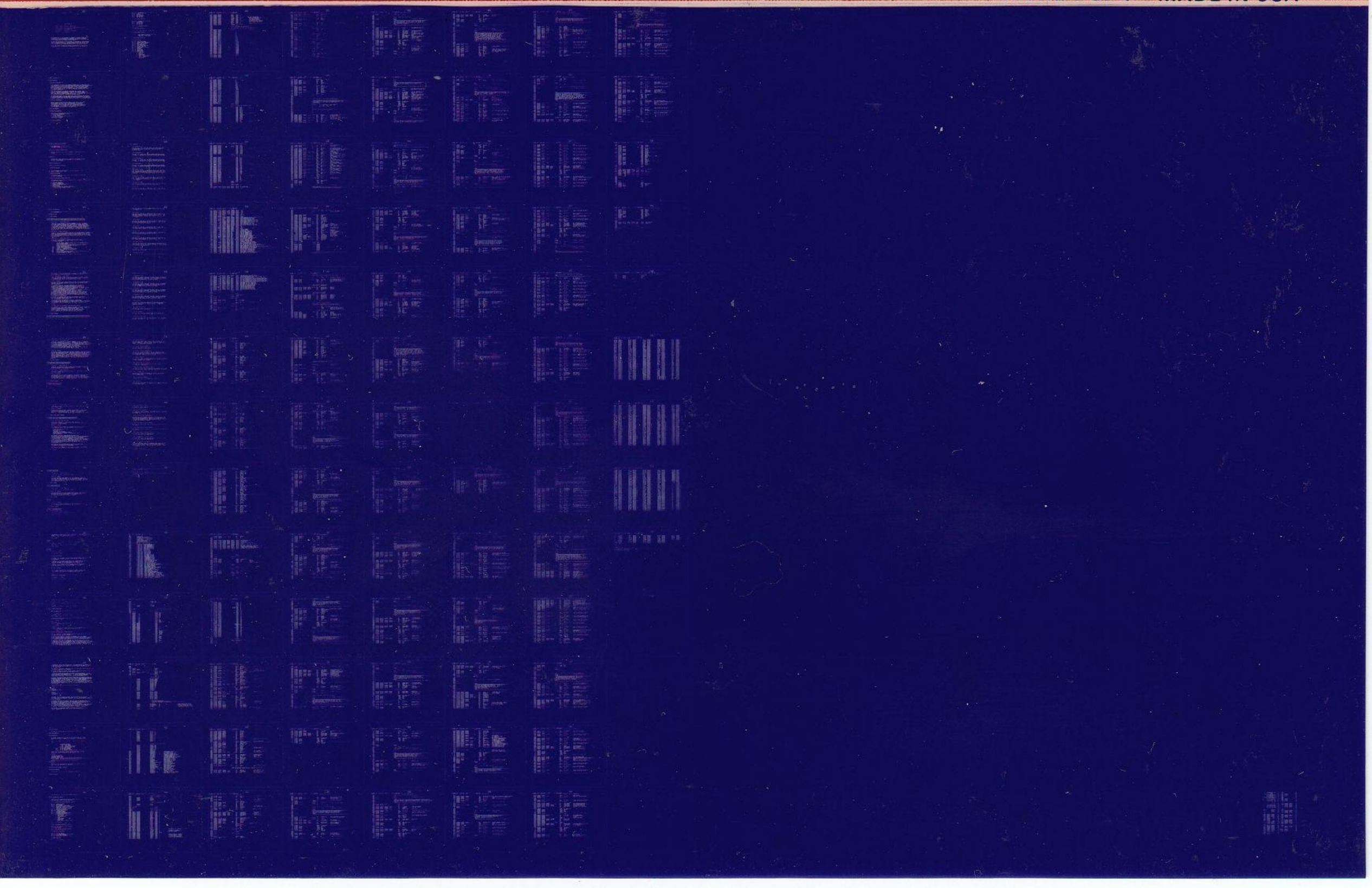
EP-DZRLA-A-DL

FICHE 1 OF 1

DEC 1977

digital

MADE IN USA



IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZRLA-A-D

PRODUCT NAME: RL11/RLV11 RL01 CONTROLLER TEST (PART 1)

DATE CREATED: 28 OCTOBER 1977

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: D. DEKNIS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

1.1.1 STRUCTURE OF PROGRAM

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

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

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

1.1.2 DIAGNOSTIC INFORMATION

THE 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 CORE
CONSOLE DEVICE (LA30,LA36,VT50,ETC.)
RL11/RLV11 CONTROLLER(S)
1 - 8 RL01 DRIVES
1 - 8 RL01K CARTRIDGES WITH BAD SECTOR FILE
KW11P, KW11L (OPTIONAL)
LINEPRINTER(OPTIONAL)

1.2.2 SOFTWARE REQUIREMENTS

MAINDEC-11-DZRLA-A

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THE RL01 SUBSYSTEM SHOULD HAVE SUCCESSFULLY RUN THE FOLLOWING PROGRAM:

MD-11-DVRLA RLV11 RL01 DISKLESS TEST (RLV11 ONLY)

1.5 ASSUMPTIONS

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

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURES

2.1.1 LOADING PROCEDURES

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

2.1.2 STARTING PROCEDURES

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

2.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION

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

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

2.2 SPECIAL ENVIRONMENTS

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

2.3 PROGRAM OPTIONS

2.3.1 START COMMAND

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

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

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

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

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

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

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

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

THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0

ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED. SEE EXAMPLE AT END OF 2.3.1.

2.3.1.4 END OF PASS SWITCH (/EOP:⟨INCR⟩)

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

2.3.1.5 EFFECT OF COMMAND

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

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

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

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

EXAMPLE:

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

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

2.3.2 RESTART COMMAND

****

RES⟨TART⟩/TESTS:⟨TEST-LIST⟩/PASS:⟨PASS-CNT⟩/FLAGS:⟨FLAG-LIST⟩/UNITS:⟨UNIT-LIST⟩

****

2.3.2.1 TESTS, PASS, AND FLAGS SWITCHES

‹TEST-LIST›, ‹PASS-CNT›, AND ‹FLAG-LIST› ARE AS IN THE START COMMAND.

2.3.2.2 UNITS SWITCH (/UNITS:‹UNIT-LIST›)

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

2.3.2.3 EFFECT OF COMMAND

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

2.3.3 CONTINUE COMMAND

****
CON(TINUE)/PASS:‹PASS-CNT/FLAGS:‹FLAG-LIST›
****

2.3.3.1 PASS SWITCH (/PASS:‹PASS-CNT›)

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

2.3.3.2 FLAG SWITCH (/FLAGS:‹FLAG-LIST›)

‹FLAG-LIST› IS SAME AS IN START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

2.3.3.3 EFFECT OF COMMAND

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

2.3.4 PROCEED COMMAND

****
PRO(CEED)/FLAGS:‹FLAG-LIST›
****

2.3.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>) SEQ 0007

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

2.3.4.2 EFFECT OF COMMAND

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

2.3.5 CREATE CORE IMAGE COMMAND

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

2.3.5.1 TESTS, PASS, AND FLAGS SWITCHES

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

2.3.5.2 EFFECT OF COMMAND

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

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

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

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

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

2.3.6 ADD COMMAND

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

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

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

2.3.6.2 EFFECT OF COMMAND

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

2.3.7 DROP COMMAND

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

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

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

2.3.7.2 EFFECT OF COMMAND

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

2.3.8 PRINT COMMAND

****
PR[I]NT
****

2.3.8.1 EFFECT OF COMMAND

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

2.3.9 DISPLAY COMMAND

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

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

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

2.3.9.2 EFFECT OF COMMAND

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

2.3.10 FLAGS COMMAND

```
*******
FLAGS)
*******
```

2.3.10.1 EFFECT OF COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

2.3.11 ZFLAGS COMMAND

```
********
ZFLAGS)
********
```

2.3.11.1 EFFECT OF COMMAND

ALL FLAGS ARE CLEARED.

2.3.12 CONTROL CHARACTERS

A CONTROL C (↑C) ENTERED VIA THE CONSOLE DEVICE DURING THE EXECUTION OF A DIAGNOSTIC CAUSES A RETURN TO THE DIAGNOSTIC SUPERVISOR COMMAND MODE.

A CONTROL Z (↑Z) ENTERED WITHIN ONE OF THE THREE OPERATOR DIALOGS (HARDCORE,HARDWARE, OR SOFTWARE QUESTIONS) CAUSES TO DEFAULT VALUES TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.

A CONTROL O (↑O) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES ALL CONSOLE OUTPUT TO BE SUPPRESSED FOR THE REMAINDER OF THE DIAGNOSTIC OR UNTIL ANOTHER CONTROL O IS TYPED.

2.3.13 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, 300?
```

ANSWER WITH THE INTERRUPT VECTOR OF THE CONTROLLER.

BR LEVEL (0 5)?

ANSWER WITH THE INTERRUPT PRIORITY OF THE CONTROLLER

DRIVE (0) 0?

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

2.3.14 SOFTWARE PARAMETERS

THE FOLLOWING QUESTIONS ARE ASKED ON A START, RESTART OR CONTINUE IF THE QUESTION:

CHANGE SW?

IS ANSWERED YES(Y). THE QUESTIONS ARE:

DROP ON ERROR LIMIT (L) Y?

TO ALLOW THE UNIT TO BE DROPPED ONCE A PREDETERMINED NUMBER OF ERRORS 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

2.3.15 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

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

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

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

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

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

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

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

NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO CONSTRUCT A SET OF P-TABLES. ASSUME THAT WE HAVE 64 UNITS, AND THAT THERE ARE THREE HARDWARE PARAMETERS FOR EACH (THREE SLOTS IN THE P-TABLE, THREE HARDWARE QUESTIONS IN THE DIALOGUE). LET THE DESIRED VALUE FOR THE FIRST PARAMETER BE THE NUMBER 75 FOR ALL 64 TABLES. LET THE DESIRED VALUE FOR THE SECOND PARAMETER BE EQUAL TO THE UNIT NUMBER (1,2,3,...,64) EXCEPT FOR UNIT 50, WHICH SHOULD RECEIVE THE VALUE 49. LET THE DESIRED VALUE FOR THE THIRD PARAMETER BE THE NUMBER 76 FOR THE FIRST 20 UNITS AND THE NUMBER 77 FOR THE LAST 44 UNITS.

THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:

```
* UNITS (D)  ?  64

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

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

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

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

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

2.4 EXECUTION TIMES

ONE PASS OF THE PROGRAM TAKES APPROXIMATELY 45 SECONDS.

3.0 ERROR INFORMATION

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
```

REGISTER DESCRIPTIONS CAN BE FOUND IN SECTION 5.0.

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.

SEQ 0013

5.0 DEVICE INFORMATION TABLES

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

RLCS - CONTROL AND STATUS REGISTER (XXXXX0)

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

BIT 0 - DRIVE READY

RLBA - BUS ADDRESS REGISTER (XXXXX2)

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

RLDA - DISK ADDRESS REGISTER (XXXXX4)

FOR READ/WRITE FUNCTIONS

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

FOR SEEK FUNCTION

BIT 15 - MUST BE ZERO(0)

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

FOR GET STATUS FUNCTION

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

RLMP - MULTIPURPOSE REGISTER

FOR READ/WRITE FUNCTION

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

FOR READ HEADER FUNCTION

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

FOR GET STATUS FUNCTION

HAS DRIVE STATUS

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

6.0 TEST SUMMARIES

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 W..N, THEN THE RLCS, RLBA AND RLDA ARE VERIFIED THAT THEY D.D 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.

SEC 0021

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.

T.C PROGRAM LISTING

```
                                .ENABLE AMA
                                .ENABLE ABS
                                .NLIST  ME,CND,MD

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


002000                                  POINTER BGNSFT,BGNSW,BGNDU,BGNAU


002000                          BGNMOD  MDHEDR

002000                                  HEADER  DZRLA,A,0
002000          104                     .ASCII  /D/
002001          132                     .ASCII  /Z/
002002          122                     .ASCII  /R/
002003          114                     .ASCII  /L/
002004          101                     .ASCII  /A/
002005          000                     .BYTE   0
002006          000                     .BYTE   0
002007          000                     .BYTE   0
002010          101                     .ASCII  /A/
002011          060                     .ASCII  /0/
002012          001                     .BYTE   C$REVISION
002013          006                     .BYTE   C$EDIT
002014  000000                          .WORD   0
002016  000000                          .WORD
002020  000000                          .WORD
002022  000000                          .WORD
002024  000000                          .WORD   0
002026  000000                          .WORD   0
002030  000000                          .WORD   0
002032  000000                          .WORD   0
002034  000000                          .WORD   0
002036  000000                          .WORD   0
002040  011124                          .WORD   L$DISPATCH
002042  011254                          .WORD   L$INIT
002044  012110                          .WORD   L$CLEAN
002046  023154                          .WORD   L$HARD
002050  023300                          .WORD   L$SOFT
002052  002104                          .WORD   L$DVTYP
002054  000000                          .WORD   0
002056  011100                          .WORD   L$HW
002060  011114                          .WORD   L$SW
002062  002102                          .WORD   L$DR
002064  002102                          .WORD   L$DRST
002066  000000                          .WORD   0
002070  012162                          .WORD   L$AU
002072  012156                          .WORD   L$DU
002074  000000                          .WORD   0
002076  023402                          .WORD   L$LAST


002100                          ENDMOD
```

```
                                                DEVREG
                                                .WORD   0
                                                .BLKW
                                               'DEVTYP  <RL01>
                                                .ASCIZ  @RL01@
                                                .EVEN
                                BGNMOD          GLBEQAT

                                                EQUALS
                                ;
                                ; BIT DIFINITIONS
                                ;
                                BIT15== 100000
                                BIT14== 40000
                                BIT13== 20000
                                BIT12== 10000
                                BIT11== 4000
                                BIT10== 2000
                                BIT09== 1000
                                BIT08== 400
                                BIT07== 200
                                BIT06== 100
                                BIT05== 40
                                BIT04== 20
                                BIT03== 10
                                BIT02== 4
                                BIT01== 2
                                BIT00== 1
                                .
                                BIT9==  BIT09
                                BIT8==  BIT08
                                BIT7==  BIT07
                                BIT6==  BIT06
                                BIT5==  BIT05
                                BIT4==  BIT04
                                BIT3==  BIT03
                                BIT2==  BIT02
                                BIT1==  BIT01
                                BIT0==  BIT00
                                ;
                                ; EVENT FLAG DEFINITIONS
                                ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                ;   EF16:EF01 AVAILABLE FOR PROGRAM USE
                                ;
                                EF.START==      32.                     ; START COMMAND WAS ISSUED
                                EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
                                EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
                                EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
                                EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                                .
                                EF15==  16.
                                EF15==  15.
```

```
                        000016          EF14==  14.
                        000015          EF13==  13.
                        000014          EF12==  12.
                        000013          EF11==  11.
                        000012          EF10==  10.
                        000011          EF09==  9.
                        000010          EF08==  8.
                        000007          EF07==  7
                        000006          EF06==  6
                        000005          EF05==  5
                        000004          EF04==  4
                        000003          EF03==  3
                        000002          EF02==  2
                        000001          EF01==  1
                                        ;
                                        ; PRIORITY LEVEL DEFINITIONS
                                        ;
                        000340          PRI07== 340
                        000300          PRI06== 300
                        000240          PRI05== 240
                        000200          PRI04== 200
                        000140          PRI03== 140
                        000100          PRI02== 100
                        000040          PRI01== 40
                        000000          PRI00== 0
  2834                  000001          DRDY=BIT0               ;DRIVE READY (RLCS)
  2835                  000100          INTEN=BIT6              ;INTERRUPT ENABLE (RLCS)
  2836                  100000          ERR=BIT15               ;RL11 ERROR (RLCS)
  2837                  040000          DERR=BIT14              ;RLCS DRIVE ERROR (RLCS)
  2838                  002000          OPI=BIT10               ;OPERATION INCOMPLETE (RLCS)
  2839                  000200          CRDY=BIT7               ;CONTROLLER READY (RLCS)
  2840                  000040          BA17=BIT5               ;EXTENDED ADDRESS BIT 17 (RLCS)
  2841                  000020          BA16=BIT4               ;EXTENDED ADDRESS BIT 16 (RLCS)
  2842                  020000          NXM=BIT13               ;NON-EXISTANT MEMORY (RLCS)
  2843                  000000          DS0=0                   ;DRIVE SELECT 0 (RLCS)
  2844                  000400          DS1=BIT8                ;DRIVE SELECT 1 (RLCS)
  2845                  001000          DS2=BIT9                ;DRIVE SELECT 2 (RLCS)
  2846                  001400          DS3=BIT8!BIT9           ;DRIVE SELECT 3 (RLCS)
  2847                  000000          NOOP0=0                 ;FUNCTION-NOOP(0)
  2848                  000016          NOOP7=BIT1!BIT2!BIT3    ;FUNCTION-NOOP(7)
  2849                  000002          WRCHK=BIT1              ;WRITE CHECK FUNCTION
  2850                  000004          GSTAT=BIT2              ;GET STATUS FUNCTION
  2851                  000006          SEEK=BIT2!BIT1          ;SEEK FUNCTION
  2852                  000010          RDHDR=BIT3              ;READ HEADER FUNCTION
  2853                  000012          WRITE=BIT3!BIT1         ;WRITE DATA FUNCTION
  2854                  000014          READ=BIT3!BIT2          ;READ DATA FUNCTION
  2855                  000202          GODRVR=BIT1!BIT7        ;CRDY AND DRDY
  2856                  000010          DRST=BIT3               ;DRIVE RESET (RLDA)
  2857                  000002          GSBIT=BIT1              ;GET STATUS BIT (RLDA)
  2858                  000001          MK=BIT0                 ;MARKER BIT (RLDA)
  2859                  000004          SIGN=BIT2               ;SIGN BIT (RLDA)
  2860                  000100          RHHS=BIT6               ;HEAD SELECT IN READ HEADER
  2861                  000100          STHS=BIT6               ;HEAD SELECT IN STATUS BACK
  2862                  000020          DAHS=BIT4               ;HEAD SELECT IN SEEK
  2863
```

```
                                ;OFFSET FOR HARDWARE P-TABLE

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

                                ;OFFSET FOR SOFTWARE P-TABLE

                    000000      DLT=0
                    000002      ELT=2
                    000004      SIZE=4

002112                                  ENDMOD

002112              BGNMOD      GLBDAT

                    .SBTTL      GLOBAL DATA

002112  000000      PWRFLG:     .WORD   0
002114  000000      UUT:        .WORD   0
002116  000000      UNITST:     .WORD   0
002120  000000      RLCS:       .WORD   0
002122  000000      RLBA:       .WORD   0
002124  000000      RLDA:       .WORD   0
002126  000000      RLMP:       .WORD   0
002130  000000      BCSR:       .WORD   0
002132  000000      BPRIOR:     .WORD   0
002134  000000      BVEC:       .WORD   0
002136  000000      DRIVE:      .WORD   0           ;DRIVE UNDER TEST
002140  000000      B.CS:       .WORD   0
002142  000000      B.BA:       .WORD   0
002144  000000      B.DA:       .WORD   0
002146  000000      B.MP:       .WORD   0
002150  000000      DERFLG:     .WORD
002152  000000      E.CS:       .WORD   0
002154  000000      E.BA:       .WORD   0
002156  000000      E.DA:       .WORD   0
002160  000000      E.MP:       .WORD   0
002162  000000      E.MP1:      .WORD   0
002164  000000      E.MP2:      .WORD   0
002166  000000      PFLG:       .WORD   0           ;PROCESSOR TYPE  0=UNIBUS  1=Q-BUS
002170  000000      TRPFLG:     .WORD   0
002172  000000      INTFLG:     .WORD   0           ;INTERRUPT OCCURANCE FLAG
002174  000000      LDCSR:      .WORD   0           ;LOCATION TO FORM RLCS
002176  000077      SECMSK:     .WORD   77          ;MASK OUT SECTOR
002200  120001      XPOLY:      .WORD   120001
002202  000004      ERRVEC:     .WORD   4
002204  000000      BCCFBK:     .WORD   0           ;LOCATION USED BY "SIMBCC"
002206  000000      CALBCC:     .WORD   0           ;LOCATION USED BY "SIMBCC"
002210  000000      TEMP2:      .WORD   0           ;LOCATION USED BY "SIMBCC"
002212  000000      TEMP3:      .WORD   0           ;LOCATION USED BY "SIMBCC"
002214  000000      TEMP4:      .WORD   0           ;LOCATION USED BY "SIMBCC"
002216  000000      TMP0:       .WORD   0
002220  000000      TMP1:       .WORD   0
```

```
2920  002222  000000                  TMP2:    .WORD   0
2921  002224  000000                  GDDAT:   .WORD   0
2922  002226  000000                  BDDAT:   .WORD   0
2923  002230  000000                  FIRST:   .WORD   0              ;FIRST SECTOR READ
2924  002232  177700                  CYLMSK:  .WORD   177700         ;MASK CYLINDER AND HEAD SELECT
2925  002234  000050                  MXSEC1:  .WORD   40.            ;MAX SECTOR ADDRESS +1
2926  002236  000047                  MAXSEC:  .WORD   39.            ;MAX SECTOR ADDRESS
2927  002240  000000                  DWORD:   .WORD   0              ;DIFFERENCE WORD (SEEK)
2928  002242  077600                  MAXCYL:  .WORD   77600          ;MAXIMUM CYLINDER ADDRESS
2929  002244  000000                  SVHD:    .WORD   0              ;SAVE CURRENT HEAD SELECT
2930  002246  000000                  ERRLMT:  .WORD   0
2931  002250  000000                  WHY:     .WORD   0              ;REASON FOR DROP IN AUTOSIZE
2932  002252  000000                  T.CNTLR:         .WORD   0
2933  002254  000000                  TMPFNC:  .WORD   0
2934
2935                                  .SBTTL  PATTERNS FOR REGISTER R/W
2936                                  ;
2937                                  ;PATTERNS USED FOR LOADING/READING REGISTERS
2938
2939  002256  000000                  BEGPAT:  0                      ;GROWING 1
2940  002260  000001                           1
2941  002262  000003                           3
2942  002264  000007                           7
2943  002266  000017                           17
2944  002270  000037                           37
2945  002272  000077                           77
2946  002274  000177                           177
2947  002276  000377                           377
2948  002300  000777                           777
2949  002302  001777                           1777
2950  002304  003777                           3777
2951  002306  007777                           7777
2952  002310  017777                           17777
2953  002312  037777                           37777
2954  002314  077777                           77777
2955  002316  177777                           177777
2956  002320  177776                           177776                 ;GROWING 0
2957  002322  177774                           177774
2958  002324  177770                           177770
2959  002326  177760                           177760
2960  002330  177740                           177740
2961  002332  177700                           177700
2962  002334  177600                           177600
2963  002336  177400                           177400
2964  002340  177000                           177000
2965  002342  176000                           176000
2966  002344  174000                           174000
2967  002346  170000                           170000
2968  002350  160000                           160000
2969  002352  140000                           140000
2970  002354  100000                           100000
2971
2972  002356  000000                           000000
2973  002360  000001                           1                      ;WALKING 1
2974  002362  000002                           2
2975  002364  000004                           4
```

```
2976  002366  000010                  10
2977  002370  000020                  20
2978  002372  000040                  40
2979  002374  000100                  100
2980  002376  000200                  200
2981  002400  000400                  400
2982  002402  001000                  1000
2983  002404  002000                  2000
2984  002406  004000                  4000
2985  002410  010000                  10000
2986  002412  020000                  20000
2987  002414  040000                  40000
2988  002416  100000                  100000
2989  002420  177777                  177777                  ;WALKING 0
2990  002422  177776                  177776
2991  002424  177775                  177775
2992  002426  177773                  177773
2993  002430  177767                  177767
2994  002432  177757                  177757
2995  002434  177737                  177737
2996  002436  177677                  177677
2997  002440  177577                  177577
2998  002442  177377                  177377
2999  002444  176777                  176777
3000  002446  175777                  175777
3001  002450  173777                  173777
3002  002452  167777                  167777
3003  002454  157777                  157777
3004  002456  137777                  137777
3005  002460  077777                  077777
3006  002462  177777                  177777
3007  002464  000000          ENDPAT: 000000
3008
3009                          .SBTTL  PATTERNS FOR DIFFERENCE WORD
3010
3011  002466  000200          SKLST:  .WORD   BIT7
3012  002470  000400                  .WORD   BIT8            ;SHIFTING 1
3013  002472  001000                  .WORD   BIT9
3014  002474  002000                  .WORD   BIT10
3015  002476  004000                  .WORD   BIT11
3016  002500  010000                  .WORD   BIT12
3017  002502  020000                  .WORD   BIT13
3018  002504  040000                  .WORD   BIT14
3019  002506  077600                  .WORD   77600           ;SHIFTING 0
3020  002510  077400                  .WORD   77400
3021  002512  076600                  .WORD   76600
3022  002514  075600                  .WORD   75600
3023  002516  073600                  .WORD   73600
3024  002520  067600                  .WORD   67600
3025  002522  057600                  .WORD   57600
3026  002524  037600                  .WORD   37600
3027  002526  077600                  .WORD   77600
3028  002530  000200                  .WORD   200
3029  002532  000600                  .WORD   600             ;GROWING 1
3030  002534  001600                  .WORD   1600
3031  002536  003600                  .WORD   3600
```

```
3032  002540  007600                          .WORD   7600
3033  002542  017600          QUAMAX:         .WORD   17600
3034  002544  037600          HALMAX:         .WORD   37600
3035  002546  077600                          .WORD   77600
3036  002550  077400                          .WORD   77400          ;GROWING 0
3037  002552  077000                          .WORD   77000
3038  002554  076000                          .WORD   76000
3039  002556  074000                          .WORD   74000
3040  002560  070000                          .WORD   70000
3041  002562  060000                          .WORD   60000
3042  002564  040000                          .WORD   40000
3043  002566  000000          SKEND:          .WORD   00000
3044
3045                          ;PATTERNS FOR TEST OF RLCS
3046
3047  002570  000000          CSPAT:          .WORD   0              ;SHIFTING 1
3048  002572  000002                          .WORD   BIT1
3049  002574  000004                          .WORD   BIT2
3050  002576  000010                          .WORD   BIT3
3051  002600  000020                          .WORD   BIT4
3052  002602  000040                          .WORD   BIT5
3053  002604  000100                          .WORD   BIT6
3054  002606  000400                          .WORD   BIT8
3055  002610  001000                          .WORD   BIT9
3056  002612  001576                          .WORD   1576           ;GROWING 0
3057  002614  001574                          .WORD   1574
3058  002616  001570                          .WORD   1570
3059  002620  001560                          .WORD   1560
3060  002622  001540                          .WORD   1540
3061  002624  001500                          .WORD   1500
3062  002626  001400                          .WORD   1400
3063  002630  001576                          .WORD   1576           ;SHIFT 0
3064  002632  001574                          .WORD   1574
3065  002634  001566                          .WORD   1566
3066  002636  001556                          .WORD   1556
3067  002640  001536                          .WORD   1536
3068  002642  001436                          .WORD   1436
3069  002644  001136                          .WORD   1136
3070  002646  000076                          .WORD   76
3071  002650  000006                          .WORD   6              ;GROWING 1
3072  002652  000016                          .WORD   16
3073  002654  000036                          .WORD   36
3074  002656  000076                          .WORD   76
3075  002660  000176                          .WORD   176            ;
3076  002662  000576                          .WORD   576
3077  002664  001576                          .WORD   1576
3078  002666  000000          CSEND:          .WORD   0
3079  002670  000240          HDRBUF:         .BLKW   160.
3080  003370                  ENDMOD
3081
3082  003370                  BGNMOD  GLBTXT
3083                          .SBTTL  GLOBAL TEXT
3084
3088  003370  047516  041440  047117  NORES:  .ASCIZ  /NO CONTROLLER/
3089  003406  047516  042040  044522  NODRY:  .ASCIZ  /NO DRIVE CONNECTED/
3090  003431     040  051104  000126  DEMES:  .ASCIZ  / DRV/
```

```
3091  003436  047040  046530  000   NXMMES: .ASCIZ  / NXM/
3092  003443  040     050117  000111 OPIMES: .ASCIZ  / OPI/
3093  003450  044040  051103  000103 HCRCMES:.ASCIZ  / HCRC/
3094  003456  044040  043116  000   HNFMES: .ASCIZ  / HNF/
3095  003463  040     041504  000113 DCKMES: .ASCIZ  / DCk/
3096  003470  042040  052114  000   DLTMES: .ASCIZ  / DLT/
3097  003475  015     000012        MSCRLF: .ASCIZ  <15><12>
3098  003500  000015                LF:     .ASCIZ  <15>
3099  003502  041440  046517  000120 COMP:   .ASCIZ  / COMP/
3100  003510  047506  041522  042105 OPIERR: .ASCIZ  /FORCED OPI(GET STATUS) CAUSED OTHER ERRORS/
3101  003563  116     047517  020120 NOPMES: .ASCIZ  /NOOP OPERATION-FLAG MODE/
3102  003614  047516  050117  047440 NOPINT: .ASCIZ  /NOOP OPERATION-INTR. MODE/
3103  003646  051127  052111  020105 WCKM   : .ASCIZ  /WRITE CHECK OPERATION-FLAG MODE/
3104  003706  051127  052111  020105 WCKINT: .ASCIZ  /WRITE CHECK OPERATION-INTR. MODE/
3105  003747  122     040505  020104 RHDMES: .ASCIZ  /READ HEADER OPERATION-FLAG MODE/
3106  004007  122     040505  020104 RHDINT: .ASCIZ  /READ HEADER OPERATION-INTR. MODE/
3107  004050  042523  045505  047440 SEKMES: .ASCIZ  /SEEK OPERATION-FLAG MODE/
3108  004101  123     042505  020113 SEKINT: .ASCIZ  /SEEK OPERATION-INTR. MODE/
3109  004133  107     052105  051440 GSTMES: .ASCIZ  /GET STATUS OPERATION-FLAG MODE/
3110  004172  042507  020124  052123 GSTINT: .ASCIZ  /GET STATUS OPERATION-INTR MODE/
3111  004231  103     035123  000040 ARLCS:  .ASCIZ  /CS: /
3112  004236  041040  035101  000040 ARLBA:  .ASCIZ  / BA: /
3113  004244  042040  035101  000040 ARLDA:  .ASCIZ  / DA: /
3114  004252  046440  035120  000040 ARLMP:  .ASCIZ  / MP: /
3115  004260  042502  047506  042522 BEREG:  .ASCIZ  /BEFORE COMMAND: /
3116  004301  124     046511  020105 AFREG:  .ASCIZ  /TIME OF ERROR:  /
3117  004322  047503  052116  047522 CRTIM:  .ASCIZ  /CONTROLLER TIMED OUT/
3118  004347  104     044522  042526 DRTIM:  .ASCIZ  /DRIVE READY TIMED OUT/
3119  004375  103     047101  047040 EM1:    .ASCIZ  /CAN NOT ADDRESS RLCS/
3120  004422  040503  020116  047516 EM2:    .ASCIZ  /CAN NOT ADDRESS RLBA/
3121  004447  103     047101  047040 EM3:    .ASCIZ  /CAN NOT ADDRESS RLDA/
3122  004474  040503  020116  047516 EM4:    .ASCIZ  /CAN NOT ADDRESS RLMP/
3123  004521  122     041514  020123 EM5:    .ASCIZ  %RLCS READ/WRITE ERROR (BIT 0 DON'T CARE.%
3124  004572  046122  040502  051040 EM6:    .ASCIZ  %RLBA READ/WRITE ERROR%
3125  004620  046122  040504  051040 EM7:    .ASCIZ  %RLDA READ/WRITE ERROR%
3126  004646  050117  020111  047527 EM11:   .ASCIZ  /OPI WOULD NOT GENERATE INTERRUPT/
3127  004707  116     020117  047111 EM13:   .ASCIZ  /NO INTERRUPT FROM NOCP(0)/
3128  004741  116     047517  024120 EM14:   .ASCIZ  /NOOP(0) MODIFIED RLMP/
3129  004767  116     047517  024120 EM15:   .ASCIZ  /NOOP(0) MODIFIED RLBA/
3130  005015  116     047517  024120 EM16:   .ASCIZ  /NOOP(0) MODIFIED RLDA/
3131  005043  111     052116  051105 EM17:   .ASCIZ  /INTERRUPT PRIORITY FAILURE/
3132  005076  042507  020124  052123 EM30:   .ASCIZ  /GET STATUS WOULD NOT INTERRUPT/
3133  005135  122     046514  020120 EM32:   .ASCIZ  /RLMP CONTAINED WRONG STATUS/
3134  005171  117     044520  042040 EM33:   .ASCIZ  /OPI DID NOT SET-GSTAT WITHOUT GS BIT/
3135  005236  050117  020111  044504 EM34:   .ASCIZ  /OPI DID NOT SET-GSTAT WITHOUT GS AND Mk BITS
3136  005313  122     040505  020104 EM37:   .ASCIZ  /READ HEADER WOULD NOT INTERRUPT/
3137  005353  102     042101  041440 EM41:   .ASCIZ  /BAD CYLINDER OR HEAD SELECT IN REPEATED READ HEADER TEST
3138  005444  040502  020104  042510 EM42:   .ASCIZ  /BAD HEADER CRC ON READ HEADER/
3139  005502  042523  052103  051117 EM43:   .ASCIZ  /SECTOR ADDRESS OUT OF SEQUENCE DURING CONSECUTIVE READ HEADERS
3140  005601  127     044522  044524 EM44:   .ASCIZ  /WRITING RLMP MODIFIED RLCS/
3141  005634  051127  052111  047111 EM45:   .ASCIZ  /WRITING RLMP MODIFIED RLBA/
3142  005667  127     044522  044524 EM46:   .ASCIZ  /WRITING RLMP MODIFIED RLDA/
3143  005722  042523  045505  053440 EM47:   .ASCIZ  /SEEK WOULD NOT INTERRUPT/
3144  005753  104     044522  042526 EM52:   .ASCIZ  /DRIVE READY CAUSED EXTRANEOUS INTERRUPT
3145  006023  102     042101  051440 EM54:   .ASCIZ  /BAD SEEK-TEST OF DIFFENCE WORD/
3146  006062  040502  020104  042510 EM55:   .ASCIZ  /BAD HEAD SELECT VIA RD HDR
```

```
SYSERR  MACY11 30.1046)  30-OCT-77  16:51  PAGE 83-8
CERLAA.P11    05-OCT-77 10:41          GLOBAL TEXT                                    SEQ 0031

 3147  006115     102  042101  044040  EM56:   .ASCIZ  /BAD HEAD SELECT VIA GET STATUS/
 3148  006154  047514  042101  047111  EM57:   .ASCII  /LOADING RLDA BEFORE DRIVE READY ON SEEK/<15><12>
 3149  006225     104  044522  042526          .ASCIZ  /DRIVE READY DID NOT SET/
 3150  006255     102  052111  051440  EM61:   .ASCIZ  /BIT SET INSTRUCTION ON RLCS YIELDED WRONG RESULT/
 3151  006336  044502  020124  046103  EM62:   .ASCIZ  /BIT CLEAR INSTRUCTION ON RLCS YIELDED WRONG RESULT/
 3152  006421     102  052111  051440  EM63:   .ASCIZ  /BIT SET INSTRUCTION ON RLBA YIELDED WRONG RESULT/
 3153  006502  044502  020124  046103  EM64:   .ASCIZ  /BIT CLEAR INSTRUCTION ON RLBA YIELDED WRONG RESULT/
 3154  006565     102  052111  051440  EM65:   .ASCIZ  /BIT SET INSTRUCTION ON RLDA YIELDED WRONG RESULT/
 3155  006646  044502  020124  046103  EM66:   .ASCIZ  /BIT CLEAR INSTRUCTION ON RLDA YIELDED WRONG RESULT/
 3156  006731     102  051525  051040  EM67:   .ASCIZ  /BUS RESET DID NOT CLEAR RLCS/
 3157  006766  052502  020123  042522  EM70:   .ASCIZ  /BUS RESET DID NOT CLEAR RLBA/
 3158  007023     102  051525  051040  EM71:   .ASCIZ  /BUS RESET DID NOT CLEAR RLDA/
 3159  007060  051127  052111  047111  EM72:   .ASCIZ  /WRITING RLCS MODIFIED RLBA/
 3160  007113     127  044522  044524  EM73:   .ASCIZ  /WRITING RLCS MODIFIED RLDA/
 3161  007146  051127  052111  047111  EM74:   .ASCIZ  /WRITING RLBA MODIFED RLCS/
 3162  007200  051127  052111  047111  EM75:   .ASCIZ  /WRITING RLBA MODIFED RLDA/
 3163  007232  051127  052111  047111  EM76:   .ASCIZ  /WRITING RLDA MODIFIED RLCS/
 3164  007265     127  044522  044524  EM77:   .ASCIZ  /WRITING RLDA MODIFIED RLBA/
 3165  007320  046122  051503  041440  EM101:  .ASCIZ  /RLCS CONTAINED FOLLOWING ERROR(S.:  /
 3166  007365  000170                  EM102:  .BLKB   120.
 3167
 3168          007556                  .EVEN
 3169
 3173  007556                          ENDMOD
 3174
 3175                                  .SBTTL  GLOBAL ERRORS
 3176
 3177  007556                          BGNMOD  GLBERR
 3178
 3179  007556                          BGNMSG  ERR0
 3180
 3181  007556  004737  010102                  JSR     PC,LINE1
 3182  007562  004737  010136                  JSR     PC,LINE2
 3183
 3184  007566  004537  012166                  JSR     R5,CKERLT       ;CHECK ERROR LIMIT
 3185  007572                                  ENDMSG
   (3) 007572                          L10000:
   (3) 007572  104023                          EMT     C$MSG
 3186
 3187  007574                          BGNMSG  ERR1
 3188
 3189  007574  004737  010102                  JSR     PC,LINE1
 3190
 3191  007600  004537  012166                  JSR     R5,CKERLT       ;CHECK ERROR LIMIT
 3192  007604                                  ENDMSG
   (3) 007604                          L10001:
   (3) 007604  104023                          EMT     C$MSG
```

```
3194
3195   007606                                  BGNMSG  ERR2
3196
3197   007606  004737  010102                          JSR     PC,LINE1
3198   007612                                          PRINTB  #FRMT4,GDDAT,BDDAT
 (9)   007612  013746  002226                          MOV     BDDAT,-(SP)
 (8)   007616  013746  002224                          MOV     GDDAT,-(SP)
 (7)   007622  012746  010560                          MOV     #FRMT4,-(SP)
 (6)   007626  012746  000003                          MOV     #3,-(SP)
 (3)   007632  010600                                  MOV     SP,R0
 (4)   007634  104014                                  EMT     C$PNTB
 (4)   007636  062706  000010                          ADD     #10,SP
3199
3200   007642  004537  012166                          JSR     R5,CKERLT       ;CHECK ERROR LIMIT
3201   007646                                          ENDMSG
 (3)   007646                                  L10002:
 (3)   007646  104023                                  EMT     C$MSG
3202
3203   007650                                  BGNMSG  ERR3
3204
3205   007650  004737  010102                          JSR     PC,LINE1
3206   007654  004737  010136                          JSR     PC,LINE2
3207   007660                                          PRINTB  #FRMT5,TMP0,BDDAT,GDDAT
 (10)  007660  013746  002224                          MOV     GDDAT,-(SP)
 (9)   007664  013746  002226                          MOV     BDDAT,-(SP)
 (8)   007670  013746  002216                          MOV     TMP0,-(SP)
 (7)   007674  012746  010616                          MOV     #FRMT5,-(SP)
 (6)   007700  012746  000004                          MOV     #4,-(SP)
 (3)   007704  010600                                  MOV     SP,R0
 (4)   007706  104014                                  EMT     C$PNTB
 (4)   007710  062706  000012                          ADD     #12,SP
3208
3209   007714  004537  012166                          JSR     R5,CKERLT       ;CHECK ERROR LIMIT
3210   007720                                          ENDMSG
 (3)   007720                                  L10003:
 (3)   007720  104023                                  EMT     C$MSG
3211
3212   007722                                  BGNMSG  ERR4
3213
3214   007722  004737  010102                          JSR     PC,LINE1
3215   007726  004737  010136                          JSR     PC,LINE2
3216   007732                                          PRINTB  #FRMT4,GDDAT,BDDAT
 (9)   007732  013746  002226                          MOV     BDDAT,-(SP)
 (8)   007736  013746  002224                          MOV     GDDAT,-(SP)
 (7)   007742  012746  010560                          MOV     #FRMT4,-(SP)
 (6)   007746  012746  000003                          MOV     #3,-(SP)
 (3)   007752  010600                                  MOV     SP,R0
 (4)   007754  104014                                  EMT     C$PNTB
 (4)   007756  062706  000010                          ADD     #10,SP
3217
3218   007762  004537  012166                          JSR     R5,CKERLT       ;CHECK ERROR LIMIT
3219   007766                                          ENDMSG
 (3)   007766                                  L10004:
 (3)   007766  104023                                  EMT     C$MSG
3220
3221   007770                                  BGNMSG  ERR5
```

```
                                                        JSR     PC,LINE1
007770  004737  010102
                                                        JSR     R5,CKERLT       ;CHECK ERROR LIMIT
007774  004537  012166
                                                        ENDMSG
010000
010000                                          L10005:
010000  104023                                          EMT     C$MSG

010002                                          BGNMSG  ERR6

010002  004737  010102                                  JSR     PC,LINE1
010006  004737  010360                                  JSR     PC,LINE3
010012  004737  010136                                  JSR     PC,LINE2


010016                                          1$:     PRINTB  #FRMT99
010016  012746  010613                                  MOV     #FRMT99,-(SP)
010022  012746  000001                                  MOV     #1,-(SP)
010026  010600                                          MOV     SP,R0
010030  104014                                          EMT     C$PNTB
010032  062706  000004                                  ADD     #4,SP
010036  004537  012166                                  JSR     R5,CKERLT       ;CHECK ERROR LIMIT
010042                                                  ENDMSG
010042                                          L10006:
010042  104023                                          EMT     C$MSG

010044                                          BGNMSG  ERR7

010044  004737  010102                                  JSR     PC,LINE1
010050                                                  PRINTB  #FRMT6,BDDAT
010050  013746  002226                                  MOV     BDDAT,-(SP)
010054  012746  010667                                  MOV     #FRMT6,-(SP)
010060  012746  000002                                  MOV     #2,-(SP)
010064  010600                                          MOV     SP,R0
010066  104014                                          EMT     C$PNTB
010070  062706  000006                                  ADD     #6,SP

010074  004537  012166                                  JSR     R5,CKERLT

010100                                                  ENDMSG
010100                                          L10007:
010100  104023                                          EMT     C$MSG

010102                                          LINE1:  PRINTB  #FRMT1,RLCS,B,DRIVE+1
010102  005046                                          CLR     -(SP)
010104  153716  002137                                  BISB    DRIVE+1,(SP)
010110  013746  002120                                  MOV     RLCS,-(SP)
010114  012746  010432                                  MOV     #FRMT1,-(SP)
010120  012746  000003                                  MOV     #3,-(SP)
010124  010600                                          MOV     SP,R0
010126  104014                                          EMT     C$PNTB
010130  062706  000010                                  ADD     #10,SP
010134  000207                                          RTS     PC

010136                                          LINE2:  PRINTB  #FRMT2,#BEREG,#ARLCS,B.CS,#ARLBA,B.BA
010136  013746  002142                                  MOV     B.BA,-(SP)
```

```
                010142  012746  004236          MOV     #ARLBA,-(SP)
                010146  013746  002140          MOV     B.CS,-(SP)
                010152  012746  004231          MOV     #ARLCS,-(SP)
                010156  012746  004260          MOV     #BEREG,-(SP)
                010162  012746  010472          MOV     #FRMT2,-(SP)
                010166  012746  000006          MOV     #6,-(SP)
                010172  010600                  MOV     SP,R0
                010174  104014                  EMT     C$PNTB
                010176  062706  000016          ADD     #16,SP
3252            010202                          PRINTB  #FRMT2A,#ARLDA,B.DA,#ARLMP,B.MP
                010202  013746  002146          MOV     B.MP,-(SP)
                010206  012746  004252          MOV     #ARLMP,-(SP)
                010212  013746  002144          MOV     B.DA,-(SP)
                010216  012746  004244          MOV     #ARLDA,-(SP)
                010222  012746  010511          MOV     #FRMT2A,-(SP)
                010226  012746  000005          MOV     #5,-(SP)
                010232  010600                  MOV     SP,R0
                010234  104014                  EMT     C$PNTB
                010236  062706  000014          ADD     #14,SP
3253            010242                          PRINTB  #FRMT2,#AFREG,#ARLCS,E.CS,#ARLBA,E.BA
                010242  013746  002154          MOV     E.BA,-(SP)
                010246  012746  004236          MOV     #ARLBA,-(SP)
                010252  013746  002152          MOV     E.CS,-(SP)
                010256  012746  004231          MOV     #ARLCS,-(SP)
                010262  012746  004301          MOV     #AFREG,-(SP)
                010266  012746  010472          MOV     #FRMT2,-(SP)
                010272  012746  000006          MOV     #6,-(SP)
                010276  010600                  MOV     SP,R0
                010300  104014                  EMT     C$PNTB
                010302  062706  000016          ADD     #16,SP
3254            010306                          PRINTB  #FRMT2B,#ARLDA,E.DA,#ARLMP,E.MP,E.MP1,E.MP2
                010306  013746  002164          MOV     E.MP2,-(SP)
                010312  013746  002162          MOV     E.MP1,-(SP)
                010316  013746  002160          MOV     E.MP,-(SP)
                010322  012746  004252          MOV     #ARLMP,-(SP)
                010326  013746  002156          MOV     E.DA,-(SP)
                010332  012746  004244          MOV     #ARLDA,-(SP)
                010336  012746  010524          MOV     #FRMT2B,-(SP)
                010342  012746  000007          MOV     #7,-(SP)
                010346  010600                  MOV     SP,R0
                010350  104014                  EMT     C$PNTB
                010352  062706  000020          ADD     #20,SP
3255            010356  000207                  RTS     PC
3256
3257            010360                  LINE3:  PRINTB  #FRMT3,#EM101
                010360  012746  007320          MOV     #EM101,-(SP)
                010364  012746  010553          MOV     #FRMT3,-(SP)
                010370  012746  000002          MOV     #2,-(SP)
                010374  010600                  MOV     SP,R0
                010376  104014                  EMT     C$PNTB
                010400  062706  000006          ADD     #6,SP
3258            010404                          PRINTB  #FRMT3,#EM102
                010404  012746  007365          MOV     #EM102,-(SP)
                010410  012746  010553          MOV     #FRMT3,-(SP)
                010414  012746  000002          MOV     #2,-(SP)
                010420  010600                  MOV     SP,R0
```

```
                                                EMT     C$PNTB
                                                ADD     #6,SP
3259    010430  000207                          RTS     PC
3260
3264
3265    010432  040445  047503  052116  FRMT1:  .ASCIZ  /%ACONTROLLER: %06%A  DRIVE: %01/
3266    010472  047045  052045  052045  FRMT2:  .ASCIZ  /%N%T%T%06%T%06/
3267    010511      045 022524  033117  FRMT2A: .ASCIZ  /%T%06%T%06/
3268    010524  052045  047445  022466  FRMT2B: .ASCIZ  /%T%06%T%06%A %06%A %06/
3269    010553      045 022516  000124  FRMT3:  .ASCIZ  /%N%T/
3270    010560  047045  040445  054105  FRMT4:  .ASCII  /%N%AEXP'D: %06%A REC'D: %06/
3271    010613      045 000116          FRMT99: .ASCIZ  /%N/
3272    010616  047045  040445  040514  FRMT5:  .ASCIZ  /%N%ALAST: %06%A PRES: %06%A EXP'D: %06%N/
3273    010667      045 022516  040501  FRMT6:  .ASCIZ  /%N%AAT PROCESSOR LEVEL %06%N/
3274    010724  040445  051105  047522  FRMT11: .ASCIZ  /%AERROR LIMIT EXCEEDED-DROPPED%N/
3275    010765      045 022516  042101  FRMT12: .ASCIZ  /%N%ADRIVE DID NOT RECOVER FROM POWER FAILURE%N
3276    011044  047045  052045  040445  FRMT13: .ASCIZ  /%N%T%A - WILL NOT TEST%N/
3277
3278            011076                  .EVEN
3279
3280
3284
3285
3286
3287    011076                          ENDMOD
3288
3289    011076                  BGNMOD  HPTCODE
3290
3291    011076                  BGNHW
 (3)    011076  000005                  .WORD   L10010-L$HW/2
3292    011100  174400                  .WORD   174400          ;CSR
3293    011102  000330                  .WORD   330             ;VECTOR
3294    011104  000240                  .WORD   240             ;PRIORITY
3295    011106  000000                  .WORD   0               ;DRIVE (BITS 8,9,10)
3296    011110  000001                  .WORD   1               ;RL11 = 1, RLV11 = 0
3297
3298    011112                  ENDHW
 (3)    011112                  L10010:
3299
3300    011112                          ENDMOD
3301
3302    011112                  BGNMOD  SPTCODE
3303
3304    011112                  BGNSW
 (3)    011112  000003                  .WORD   L10011-L$SW/2
3305
3306    011114  000000          DROP:   .WORD   0
3307    011116  000012          MERLMT: .WORD   10.
3308    011120  000000          T.SIZE: .WORD   0
3309
3310    011122                  ENDSW
 (3)    011122                  L10011:
3311
3312    011122                          ENDMOD
3313
3314    011122                  BGNMOD  DSPCODE
```

```
 3315
 3316  011122                          DISPATCH        44
  (4)  011122  000054                            .WORD   44
  (6)  011124  013440                            .WORD   T1
  (6)  011126  013534                            .WORD   T2
  (6)  011130  013630                            .WORD   T3
  (6)  011132  013724                            .WORD   T4
  (6)  011134  014020                            .WORD   T5
  (6)  011136  014140                            .WORD   T6
  (6)  011140  014242                            .WORD   T7
  (6)  011142  014330                            .WORD   T8
  (6)  011144  014454                            .WORD   T9
  (6)  011146  014600                            .WORD   T10
  (6)  011150  014704                            .WORD   T11
  (6)  011152  015004                            .WORD   T12
  (6)  011154  015074                            .WORD   T13
  (6)  011156  015174                            .WORD   T14
  (6)  011160  015304                            .WORD   T15
  (6)  011162  015356                            .WORD   T16
  (6)  011164  015414                            .WORD   T17
  (6)  011166  015540                            .WORD   T18
  (6)  011170  015700                            .WORD   T19
  (6)  011172  016040                            .WORD   T20
  (6)  011174  016244                            .WORD   T21
  (6)  011176  016274                            .WORD   T22
  (6)  011200  016500                            .WORD   T23
  (6)  011202  016564                            .WORD   T24
  (6)  011204  016730                            .WORD   T25
  (6)  011206  016760                            .WORD   T26
  (6)  011210  017046                            .WORD   T27
  (6)  011212  017134                            .WORD   T28
  (6)  011214  017262                            .WORD   T29
  (6)  011216  017304                            .WORD   T30
  (6)  011220  017364                            .WORD   T31
  (6)  011222  017530                            .WORD   T32
  (5)  011224  017666                            .WORD   T33
  (6)  011226  020204                            .WORD   T34
  (6)  011230  020242                            .WORD   T35
  (6)  011232  020306                            .WORD   T36
  (6)  011234  020432                            .WORD   T37
  (6)  011236  021026                            .WORD   T38
  (6)  011240  021160                            .WORD   T39
  (5)  011242  021322                            .WORD   T40
  (6)  011244  021462                            .WORD   T41
  (6)  011246  021634                            .WORD   T42
  (6)  011250  022162                            .WORD   T43
  (6)  011252  022660                            .WORD   T44
 3317
 3318  011254                                    ENDMOD
 3319
 3320                                  .SBTTL  INITIALIZATION CODE
 3321  011254                          BGNMOD  INITCODE
 3322
 3323  011254                                  BGNINIT
 3324
 3325  011254                                  BRESET
```

```
                                                EMT     C$RESET
                                                READEF  #EF.PWR                 ;POWER UP?????
                                                MOV     #EF.PWR,R0
                                                EMT     C$REFG
                                                BNCOMPLETE      NOPWR           ;NO,BRANCH
                                                BCC     NOPWR
                                                MOV     L$UNIT,PWRFLG           ;YES, SET POWER FLAG
                                                BR      CONT                    ;GO TO CONTINUE POINT
                                        NOPWR:  READEF  #EF.RESTART     ;RESTART?
                                                MOV     #EF.RESTART,R0
                                                EMT     C$REFG
                                                BCOMPLETE START
                                                BCS     START
                                                READEF  #EF.NEW                 ;NEW PASS????
                                                MOV     #EF.NEW,R0
                                                EMT     C$REFG
                                                BCOMPLETE       START           ;YES,THEN RE INIT
                                                BCS     START
                                                READEF  #EF.START               ;START???
                                                MOV     #EF.START,R0
                                                EMT     C$REFG
                                                BNCOMPLETE      CONTINUE
                                                BCC     CONTINUE
002014  002114  START:  MOV     L$UNIT,UUT
177777  002116          MOV     #-1,UNITST
                        BR      NXT

                CONTINUE:       READEF  #EF.CONTINUE    ;CONTINUE????
                                MOV     #EF.CONTINUE,R0
                                EMT     C$REFG
                                BCOMPLETE       CONT
                                BCS     CONT

002114          NXT:    TST     UUT                     ;DONE ALL UUT'S
                        BNE     1$                      ;NO
177777  002116          MOV     #-1,UNITST
002014  002114          MOV     L$UNIT,UUT

002116          1$:     INC     UNITST
002114          DEC     UUT
                REST:   GPHARD  UNITST,R0
002116          MOV     UNITST,R0
                EMT     C$GPHRD
                BCOMPLETE       1$
                BCS     1$
002112          TST     PWRFLG                  ;POWER FLAG TO 0
                BEQ     NXT                     ;YES, DONT DEC IT
002112          DEC     PWRFLG
                BR      NXT                     ;GET NEXT ONE
002130          1$:     MOV     (R0)+,BCSR
002134          MOV     (R0)+,BVEC
002132          MOV     (R0)+,BPRIOR
002136          MOV     (R0)+,DRIVE
002252          MOV     (R0)+,T.CNTLR   ;GET CONTROLLER TYPE

002246          CLR     ERRLMT                  ;INIT ERROR COUNT
```

```
3363  011462  013700  002130          CONT:   MOV     BCSR,R0
3364  011466  010037  002120                  MOV     R0,RLCS
3365  011472  062700  000002                  ADD     #2,R0
3366  011476  010037  002122                  MOV     R0,RLBA
3367  011502  062700  000002                  ADD     #2,R0
3368  011506  010037  002124                  MOV     R0,RLDA
3369  011512  062700  000002                  ADD     #2,R0
3370  011516  010037  002126                  MOV     R0,RLMP
3371  011522  005737  002112                  TST     PWRFLG
3372  011526  001064                          BNE     5$
3373  011530  005737  011120                  TST     T.SIZE                  ;DO WE WANT TO CHECK UNITS??
3374  011534  001461                          BEQ     5$                      ;NO
3375
3376  011536  005037  002170                  CLR     TRPFLG                  ;CLR OUT TRAP FLAG
3377  011542                                  SETVEC  ERRVEC,#TRPHAN,#340     ;SETUP VECTOR TO CATCH NON-EXIST
 (7)  011542  012746  000340                  MOV     #340,-(SP)
 (6)  011546  012746  013302                  MOV     #TRPHAN,-(SP)
 (5)  011552  013746  002202                  MOV     ERRVEC,-(SP)
 (4)  011556  012746  000003                  MOV     #3,-(SP)
 (3)  011562  104037                          EMT     C$SVEC
 (2)  011564  062706  000010                  ADD     #10,SP
3378  011570  005777  170324                  TST     @RLCS                   ;ACCESS CONTROLLER
3379  011574                                  CLRVEC  ERRVEC                  ;RELEASE VECTOR
 (3)  011574  013700  002202                  MOV     ERRVEC,R0
 (3)  011600  104036                          EMT     C$CVEC
3380  011602  005737  002170                  TST     TRPFLG                  ;DID IT TRAP
3381  011606  001404                          BEQ     7$                      ;NO, CHECK IT'S DRIVE
3382  011610  012737  003370  002250          MOV     #NORES,WHY              ;SETUP ERR MESS
3383  011616  000415                          BR      8$
3384
3385  011620  012777  000200  170272  7$:     MOV     #200,@RLCS              ;CONTROLLER READY
3386  011626  053777  002136  170264          BIS     DRIVE,@RLCS             ;SELECT DRIVE
3387  011634  032777  000001  170256          BIT     #1,@RLCS                ;DRIVE THERE
3388  011642  001016                          BNE     5$                      ;YES
3389  011644  012737  003406  002250          MOV     #NODRY,WHY              ;SETUP ERR MESS
3390  011652                          8$:     PRINTB  #FRMT13,WHY
  8)  011652  013746  002250                  MOV     WHY,-(SP)
 (7)  011656  012746  011044                  MOV     #FRMT13,-(SP)
 (6)  011662  012746  000002                  MOV     #2,-(SP)
 (3)  011666  010600                          MOV     SP,R0
 (4)  011670  104014                          EMT     C$PNTB
 (4)  011672  062706  000006                  ADD     #6,SP
3391  011676  000434                          BR      6$
3392
3393  011700  005737  002112          5$:     TST     PWRFLG                  ;RECENT POWER FAILURE????
3394  011704  001457                          BEQ     END                     ;NO
3395
3396                                  ;THERE WAS A RECENT POWER FAILURE, THEREFORE WE WILL WAIT
3397                                  ;SIXTY SECONDS FOR THE DRIVE TO COME READY
3398
3399  011706  012701  000074                  MOV     #60.,R1                 ;SIXTY SECOND TIMEOUT
3400  011712  012777  000200  170200          MOV     #200,@RLCS              ;SET CRDY
3401  011720  053777  002136  170172          BIS     DRIVE,@RLCS             ;SET IN DRIVE SELECT
3402  011726  032777  000001  170164  2$:     BIT     #DRDY,@RLCS             ;DRIVE READY???
3403  011734  001023                          BNE     3$                      ;YES, THEN START TEST
3404
```

```
                                                WAITMS  #10.                   ;WAIT A SECOND
3405  011736                                    MOV     #10.,R0
 (3)  011736  012700  000012                    EMT     C$WTM
 (3)  011742  104026
3406
3407  011744  005301                            DEC     R1                      ;SIXTY SECONDS GONE BY
3408  011746  001367                            BNE     2$                      ;NO. GO BACK
3409
3410  011750                                    PRINTB  #FRMT12                 ;DROPPING DRIVE
 (7)  011750  012746  010765                    MOV     #FRMT12,-(SP)
 (6)  011754  012746  000001                    MOV     #1,-(SP)
 (3)  011760  010600                            MOV     SP,R0
 (4)  011762  104014                            EMT     C$PNTB
 (4)  011764  062706  000004                    ADD     #4,SP
3411  011770  004737  010102          6$:       JSR     PC,LINE1                ;GIVE DRIVE INFO
3412  011774                                    DODU    UNITST                  ;TELL SUPERVISOR TO DROP IT
 (3)  011774  013700  002116                    MOV     UNITST,R0
 (3)  012000  104053                            EMT     C$DODU
3413  012002                                    DOCLN                           ;FORCE AN ABORT
 (3)  012002  104044                            EMT     C$DCLN
3414
3415
3416  012004  012777  000013  170112  3$:       MOV     #13,@RLDA               ;SETUP DR RST
3417  012012  012777  000204  170100            MOV     #204,@RLCS              ;GS FUNC
3418  012020  053777  002136  170072            BIS     DRIVE,@RLCS             ;SELECT DRIVE
3419  012026  042777  000200  170064            BIC     #200,@RLCS              ;ISSUE IT
3420  012034  032777  000200  170056  4$:       BIT     #200,@RLCS              ;WAIT FOR READY
3421  012042  001774                            BEQ     4$
3422
3423  012044                          END:      SETVEC  BVEC,#INTSRV,#340
 (7)  012044  012746  000340                    MOV     #340,-(SP)
 (6)  012050  012746  013310                    MOV     #INTSRV,-(SP)
 (5)  012054  013746  002134                    MOV     BVEC,-(SP)
 (4)  012060  012746  000003                    MOV     #3,-(SP)
 (3)  012064  104037                            EMT     C$$VEC
 (2)  012066  062706  000010                    ADD     #10,SP
3424  012072  005037  002166                    CLR     PFLG                    ;CLR PROCESSOR FLAG
3425  012076                                    READBUS                         ;Q-BUS
 (3)  012076  104007                            EMT     C$RDBU
3426  012100                                    BNCOMPLETE      1$
 (2)  012100  103002                            BCC     1$
3427  012102  005237  002166                    INC     PFLG                    ;NO, Q-BUS THEN
3428  012106                          1$:
3429  012106                                    ENDINIT
 (3)  012106                          L10012:
 (3)  012106  104011                            EMT     C$INIT
3430
3431  012110                                    ENDMOD
3432
3433  012110                          BGNMOD    CLNCODE
3434
3435  012110                                    BGNCLN
3436
3437  012110                                    SETPRI  #PRI07
 (3)  012110  012700  000340                    MOV     #PRI07,R0
 (3)  012114  104041                            EMT     C$SPRI
3438
```

```
OUTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 84-8
DZRLAA.P11    05-OCT-77 10:41          INITIALIZATION CODE                                                    SEQ 0040

 3439  012116  032777  000200  167774  1$:     BIT     #CRDY,@RLCS
 3440  012124  001774                          BEQ     1$
 3441
 3442  012126  042777  000100  167764          BIC     #INTEN,@RLCS
 3443
 3444  012134                                  CLRVEC  BVEC
   (3) 012134  013700  002134                  MOV     BVEC,R0
   (3) 012140  104036                          EMT     C$CVEC
 3445
 3446
 3447
 3448  012142  005737  002112                  TST     PWRFLG                  ;TREAT POWER FAILURE
 3449  012146  001402                          BEQ     2$
 3450
 3451  012150  005337  002112                  DEC     PWRFLG
 3452
 3453  012154                          2$:
 3454  012154                                  ENDCLN
   (3) 012154                          L10013:
   (3) 012154  104012                          EMT     C$CLEAN
 3455
 3456  012156                                  ENDMOD
 3457
 3458
 3459
 3460  012156                          BGNMOD  DRPCODE
 3461
 3462  012156                                  BGNDU
 3463
 3464  012156  000240                          NOP
 3465
 3466  012160                                  ENDDU
   (3) 012160                          L10014:
   (3) 012160  104055                          EMT     C$DU
 3467
 3468  012162                          ENDMOD
 3469
 3470  012162                          BGNMOD  ADDCODE
 3471
 3472  012162                                  BGNAU
 3473
 3474  012162  000240                          NOP
 3475
 3476  012164                                  ENDAU
   (3) 012164                          L10015:
   (3) 012164  104054                          EMT     C$AU
 3477
 3478  012166                          ENDMOD
 3479
 3480
 3481                                  .SBTTL  GLOBAL SUBROUTINES
 3482
 3483  012166                          BGNMOD  GLBSUB
 3484
 3485  012166                          CKERLT: INLOOP  *
   (3) 012166  104020                          EMT     C$INLP
```

```
3486  012170                                    BCOMPLETE       99$
 (2)  012170  103427                            BCS     99$
3487  012172  005737  011114                    TST     DROP
3488  012176  001424                            BEQ     99$
3489  012200  005237  002246                    INC     ERRLMT
3490  012204  023737  002246  011116            CMP     ERRLMT,MERLMT
3491  012212  002416                            BLT     99$
3492
3493  012214                                    PRINTF  #FRMT11
 (7)  012214  012746  010724                    MOV     #FRMT11,-(SP)
 (6)  012220  012746  000001                    MOV     #1,-(SP)
 (3)  012224  010600                            MOV     SP,R0
 (4)  012226  104017                            EMT     C$PNTF
 (4)  012230  062706  000004                    ADD     #4,SP
3494  012234  004737  010102                    JSR     PC,LINE1
3495  012240                                    DODU    UNITST          ;DROP THE UNIT
 (3)  012240  013700  002118                    MOV     UNITST,R0
 (3)  012244  104053                            EMT     C$DODU
3496  012246                                    DOCLN
 (3)  012246  104044                            EMT     C$DCLN
3497  012250                              99$:
3498  012250  000205                            RTS     R5
3499
3500
3501
3502                                    .SBTTL  ROUTINE TO CHECK FOR CONTROLLER ERRORS
3503
3504                                    ;*****************************************************************
3505                                    ;*THIS ROUTINE WILL CHECK RLCS FOR ERRORS AND PRINT THEM
3506                                    ;*ACCORDINGLY.  IT WILL MERGE THE ERROR PRINTOUT WITH THE TEST
3507                                    ;*ERROR MESSAGE.
3508                                    ;*
3509                                    ;*EXAMPLE:    RLCS CONTAINED FOLLOWING ERROR(S):
3510                                    ;*                DRV    OPI    HCRC    HNF
3511                                    ;*            SEEK UNDER INTERRUPT
3512                                    ;*
3513                                    ;*
3514                                    ;*
3515                                    ;*ROUTINE USES R0,R1 AND PICKS HEADER FROM R3
3516                                    ;*
3517                                    ;*      CALL    JSR     R5,CHERR
3518                                    ;*
3519                                    ;*
3520                                    ;*
3521
3522  012252  005037  002150            CHERR:  CLR     DERFLG          ;CLEAR OUT DRIVE ERROR FLAG
3523  012256  032737  176000  002152            BIT     #176000,E.CS    ;ANY ERRORS SET
3524  012264  001001                            BNE     199$            ;IF YES, INVESTIGATE
3525  012266  000205                            RTS     R5              ;NO, EXIT
3526  01227C  023727  002254  000004    199$:   CMP     TMPFNC,#GSTAT   ;FUNCTION-NOP, RESET, GETSTATUS
3527  012276  002401                            BLT     98$             ;YES, GO CHECK IF ONLY DRIVE ERROR
3528  012300  000414                            BR      1$              ;YES SERVICE ERROR
3529  012302  023727  002254  000002    98$:    CMP     TMPFNC,#WRCHK
3530  012310  001410                            BEQ     1$
3531  012312  013700  002152                    MOV     E.CS,R0         ;GET E.CS
3532  012316  042700  001777                    BIC     #1777,R0        ;
```

```
3533  012322  022700  140000            CMP     #140000,R0      ;DRIVE ERROR ALONE?
3534  012326  001001                    BNE     1$              ;NO, GO SERVICE
3535  012330  000205              2$:   RTS     R5              ;YES, EXIT
3536
3537  012332  012701  007365      1$:   MOV     #EM102,R1       ;GET START OF STRING
3538  012336  005737  002152            TST     E.CS            ;IS COMPOSITE ERROR SET?(BETTER BE)
3539  012342  100003                    BPL     99$             ;IT'S NOT SOMETHING IS WRONG
3540  012344  004537  013016            JSR     R5,FIX          ;YES, PUT "COMP" IN STRING
3541  012350  003502                    COMP                    ;"COMP"
3542  012352  032737  040000  002152  99$:   BIT     #DERR,E.CS      ;DRIVE ERROR SET?
3543  012360  001405                    BEQ     3$              ;NO, CONTINUE
3544  012362  005237  002150            INC     DERFLG          ;SET DRV ERROR FLAG
3545  012366  004537  013016            JSR     R5,FIX          ;YES, PUT "DRV" INTO STRING
3546  012372  003431                    DEMES                   ;"DRV"
3547  012374  032737  020000  002152  3$:   BIT     #NXM,E.CS       ;NON-EXISTENT MEMORY ERROR?
3548  012402  001403                    BEQ     4$              ;NO, CONTINUE
3549  012404  004537  013016            JSR     R5,FIX          ;YES, PUT "NXM" INTO STRING
3550  012410  003436                    NXMMES                  ;"NXM"
3551  012412  032737  002000  002152  4$:   BIT     #OPI,E.CS       ;IS OPI SET?
3552  012420  001422                    BEQ     6$              ;NO, GO CHECK BITS 11 & 12
3553  012422  004537  013016            JSR     R5,FIX          ;PUT "OPI" INTO STRING
3554  012426  003443                    OPIMES                  ;"OPI"
3555  012430  032737  004000  002152    BIT     #BIT11,E.CS     ;HEADERCRC ERROR?
3556  012436  001403                    BEQ     5$              ;NO, GO CHECK HEADER NOT FOUND
3557  012440  004537  013016            JSR     R5,FIX          ;GO PUT "HCRC" IN STRING
3558  012444  003450                    HCRCMES                 ;"HCRC"
3559  012446  032737  010000  002152  5$:   BIT     #BIT12,E.CS     ;HEADER NOT FOUND?
3560  012454  001422                    BEQ     8$              ;NO, GO PUT "CRLF" IN STRING
3561  012456  004537  013016            JSR     R5,FIX          ;PUT "HNF" IN STRING
3562  012462  003456                    HNFMES                  ;"HNF"
3563  012464  000416                    BR      8$              ;PUT "CRLF" IN STRING
3564  012466  032737  004000  002152  6$:   BIT     #BIT11,E.CS     ;DATA CRC ERROR?
3565  012474  001403                    BEQ     7$              ;NO, GO CHECK DATA LATE
3566  012476  004537  013016            JSR     R5,FIX          ;PUT "DCK" IN STRING
3567  012502  003463                    DCKMES                  ;"DCK"
3568  012504  032737  010000  002152  7$:   BIT     #BIT12,E.CS     ;DATA LATE ERROR?
3569  012512  001403                    BEQ     8$              ;NO, GO PUT IN "CR_F"
3570  012514  004537  013016            JSR     R5,FIX          ;PUT "DLT" IN STRING
3571  012520  003470                    DLTMES                  ;"DLT"
3572  012522  004537  013016      8$:   JSR     R5,FIX
3573  012526  003475                    MSCRLF
3574  012530  004537  013016            JSR     R5,FIX
3575  012534  000000            RESTMS: .WORD   0               ;HEADER FROM TEST
3576  012536  105011                    CLRB    (R1)            ;PUT TERMINATOR IN
3577
3578  012540                            ERRDF   300.,LF,ERR6
  (2) 012540  104462                    TRAP    T$ERCODE
  (5) 012542  000454                    .WORD   300
  (5) 012544  003500                    .WORD   LF
  (5) 012546  010002                    .WORD   ERR6
3579
3580  012550  000205                    RTS     R5              ;EXIT ROUTINE
3581
3582                            .SBTTL LOAD RLCS
3583                            ;*****************************************************************
3584                            ;* ROUTINE TO LOAD RLCS WITH FUNCTION TO BE PERFORMED
```

```
                                        ;*      CALL:   JSR     R5,LDFUNC
                                        ;*              .WORD           ;BITS TO BE LOADED, FUNCTION
                                        ;*                              ;AND INTR ENABLE ONLY
                                        ;*
                                        ;
012552  012537  002174          LDFUNC: MOV     (R5)+,LDCSR     ;GET BITS TO LOAD
012556  005737  002150                  TST     DERFLG
012562  001424                          BEQ     98$
012564  013746  002140                  MOV     B.CS,-(SP)
012570  012777  000013  167326          MOV     #13,@RLDA
012576  012737  000004  002140          MOV     #GSTAT,B.CS
012604  053737  002136  002140          BIS     DRIVE,B.CS
012612  013777  002140  167300          MOV     B.CS,@RLCS
012620  012637  002140                  MOV     (SP)+,B.CS
012624  032777  000200  167266  99$:    BIT     #200,@RLCS
012632  001774                          BEQ     99$
012634  010346                  98$:    MOV     R3,-(SP)        ;SAVE R3
012636  042737  177661  002174          BIC     #177661,LDCSR   ;CLEAR ALL BUT FUNC & INTR EN
012644  013737  002174  012770          MOV     LDCSR,FNDFNC    ;SAVE FUNCTION
012652  042737  000100  012770          BIC     #INTEN,FNDFNC   ;ONLY FUNCTION
012660  013737  012770  002254          MOV     FNDFNC,TMPFNC
012666  012703  012772                  MOV     #HDRLST,R3      ;GET HEADER LIST
012672  006237  012770                  ASR     FNDFNC          ;ALIGN TO RIGHT
012676  001404                          BEQ     2$
012700  022323                  1$:     CMP     (R3)+,(R3)+     ;BUMP R3 BY 4
012702  005337  012770                  DEC     FNDFNC          ;FOUND IT
012706  001374                          BNE     1$              ;NO,KEEP LOOKING
012710  032737  000100  002174  2$:     BIT     #INTEN,LDCSR    ;YES,DO WE WANT FLAG OR INTR
012716  001401                          BEQ     3$              ;FLAG BRANCH
012720  005723                          TST     (R3)+           ;INTR POINT TO THAT ONE
012722  011303                  3$:     MOV     (R3),R3         ;SET HEADER
012724  010337  012534                  MOV     R3,RESTMS       ;SET UP HEADER
012730  053737  002136  002174          BIS     DRIVE,LDCSR     ;SELECT DRIVE
012736  052737  000200  002174  4$:     BIS     #200,LDCSR      ;CONTROLLER READY
012744  013777  002174  167146          MOV     LDCSR,@RLCS
012752  004537  013030                  JSR     R5,BEFORE
012756  042777  000200  167134  5$:     BIC     #200,@RLCS
012764  012603                          MOV     (SP)+,R3        ;RESTORE R3
012766  000205                          RTS     R5              ;EXIT

012770  000000                  FNDFNC: .WORD   0

012772  003563                  HDRLST: NOPMES
012774  003614                          NOPINT
012776  003646                          WCKMES
013000  003706                          WCKINT
013002  004133                  OKHDR:  GSTMES
013004  004172                          GSTINT
013006  004050                          SEKMES
013010  004101                          SEKINT
013012  003747                          RHDMES
013014  004007                          RHDINT

                                        ;*********************************************************************
                                        ;*ROUTINE TO MOVE ASCII STRINGS
```

```
OUTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 84-12
DZRLAA.P11    05-OCT-77 10:41          LOAD RLCS                                SEQ 0044

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

013016  012500                 FIX:    MOV     (R5)+,R0        ;GET ADDRESS AND MOVE RETURN
013020  112021                 1$:     MOVB    (R0)+,(R1)+     ;GET BYTE AND UPDATE
013022  001376                         BNE     1$              ;WATCH 0 BYTE TERMINATOR
013024  105741                         TSTB    -(R1)           ;BACK UP OVER ZERO BYTE
013026  000205                         RTS     R5              ;EXIT


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

013030  017737  167064  002140   BEFORE: MOV     @RLCS,B.CS      ;READ CS
013036  017737  167060  002142           MOV     @RLBA,B.BA      ;READ BA
013044  017737  167054  002144           MOV     @RLDA,B.DA      ;READ DA
013052  017737  167050  002146           MOV     @RLMP,B.MP      ;READ MP
013060  000205                           RTS     R5


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

013062  017737  167032  002152   AFTER:  MOV     @RLCS,E.CS      ;READ CS
013070  017737  167026  002154           MOV     @RLBA,E.BA      ;READ BA
013076  017737  167022  002156           MOV     @RLDA,E.DA      ;READ DA
013104  017737  167016  002160           MOV     @RLMP,E.MP      ;READ MP
013112  017737  167010  002162           MOV     @RLMP,E.MP1     ;READ MP
013120  017737  167002  002164           MOV     @RLMP,E.MP2     ;READ MP
013126  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 (1-15)
                                       ;               .WORD           ;DATA FOR CRC CALCULATION
                                       ;               .WORD           ;PREVIOUS OR STARTING CRC
                                       ;                               ;(SHOULD BE ZEROED FOR START)
                                       ;       ROUTINE USES R0,R1,R2
                                       ;
013130  010046                 SIMBCC: MOV     R0,-(SP)        ;SAVE R0
013132  010146                         MOV     R1,-(SP)        ;SAVE R1
013134  010246                         MOV     R2,-(SP)        ;SAVE R2
013136  012537  002210                 MOV     (R5)+,TEMP2     ;GET NUMBER OF BITS
013142  012537  002212                 MOV     (R5)+,TEMP3     ;GET DATA FOR CRC CALCULATION
013146  012537  002214                 MOV     (R5)+,TEMP4     ;GET STARTING CRC
013152  005037  002204         1$:     CLR     BCCFBK          ;
013156  013700  002214                 MOV     TEMP4,R0        ;GET PRESENT CRC
013162  006037  002212                 ROR     TEMP3           ;ROTATE NEW DATA
```

```
3697  013166  005500                          ADC     R0              ;MERGE NEW WITH OLD
3698  013170  032700  000001                  BIT     #1,R0           ;BIT 0 SET
3699  013174  001402                          BEQ     2$              ;IF NOT CONTINUE
3700  013176  005137  002204                  COM     BCCFBK          ;
3701  013202  013700  002200          2$:     MOV     XPOLY,R0        ;GET CRC POLYNOMIAL (CRC-16.
3702  013206  005100                          COM     R0              ;COMPLIMENT POLYNOMIAL
3703  013210  040037  002204                  BIC     R0,BCCFBK
3704  013214  000241                          CLC                     ;CLEAR CARRY
3705  013216  006037  002214                  ROR     TEMP4
3706  013222  013700  002204                  MOV     BCCFBK,R0
3707  013226  013701  002214                  MOV     TEMP4,R1
3708  013232  010102                          MOV     R1,R2
3709  013234  040100                          BIC     R1,R0
3710  013236  043702  002204                  BIC     BCCFBK,R2
3711  013242  050200                          BIS     R2,R0
3712  013244  043737  002200  002214          BIC     ^POLY,TEMP4
3713  013252  050037  002214                  BIS     R0,TEMP4
3714  013256  005337  002210                  DEC     TEMP2
3715  013262  001333                          BNE     1$
3716  013264  013737  002214  002206          MOV     TEMP4,CALBCC
3717  013272  012602                          MOV     (SP)+,R2
3718  013274  012601                          MOV     (SP)+,R1
3719  013276  012600                          MOV     (SP)+,R0
3720  013300  000205                          RTS     R5              ;RETURN
3721
3722
3723
3724                                  ;ROUTINE TO SET FLAG IF TRAP OCCURRED
3725                                  ;"TRPHAN" IS IN LOCATION 4.
3726
3727
3728  013302  005237  002170          TRPHAN: INC     TRPFLG          ;INDICATE TRAP
3729  013306  000002                          RTI                     ;RETURN
3730
3731  013310                          BGNSRV
3732
3733  013310  005237  002172          INTSRV: INC     INTFLG          ;INDICATE INTERRUPT
3734
3735  013314                          ENDSRV
(3)   013314                          L10016:
(2)   013314  000002                          RTI
3736
3737                                  ;ROUTINE TO WAIT FOR DRIVE READY
3738  013316  010146                  WTDRDY: MOV     R1,-(SP)        ;SAVE R1
3739  013320  012701  003720                  MOV     #2000.,R1       ;TIME OUT OF 200 MILLISECONDS
3740  013324  032777  000001  166566  1$:     BIT     #DRDY,@RLCS     ;DRIVE READY?
3741  013332  001011                          BNE     2$              ;YES, EXIT
3742
3743  013334                                  WAITUS  #1              ;WAIT A WHILE
(3)   013334  012700  000001                  MOV     #1,R0
3)    013340  104027                          EMT     C$WTU
3744  013342  005301                          DEC     R1              ;CHECK IF TIME UP
3745  013344  001367                          BNE     1$              ;NO, GO CHECK DRIVE READY
3746
3747  013346                                  ERRDF   200.,DRTIM,ERRS ;DRIVE READY DID NOT SET
3)    013346  104462                          TRAP    T$ERCODE
```

```
(5)  013350  000310                        .WORD   200
(5)  013352  004347                        .WORD   DRTIM
(5)  013354  007770                        .WORD   ERR5
3748
3749  013356  012601                2$:    MOV     (SP)+,R1        ;RESTORE
3750  013360  000205                        RTS     R5              ;EXIT
3751
3752                                 ;ROUTINE TO WAIT FOR CONTROLLER READY
3753  013362  010146                WTCRDY: MOV     R1,-(SP)        ;SAVE R1
3754  013364  012701  017500                MOV     #8000.,R1       ;WAIT 800 MILLISECONDS
3755  013370  032777  000200  166522 1$:    BIT     #CRDY,@RLCS     ;CONTROLLER READY
3756  013376  001014                        BNE     2$              ;YES, EXIT
3757  013400                                WAITUS  #1              ;WAIT A WHILE
(3)  013400  012700  000001                MOV     #1,R0
(3)  013404  104027                        EMT     C$WTU
3758  013406  005301                        DEC     R1              ;CHECK IF TIME UP
3759  013410  001367                        BNE     1$              ;NO GO BACK
3760
3761  013412  004537  013062                JSR     R5,AFTER        ;GET REGISTERS
3762
3763  013416                                ERRDF   100.,CRTIM,ERR6 ;CONTROLLER TIMED OUT
(3)  013416  104462                        TRAP    T$ERCODE
(5)  013420  000144                        .WORD   100
(5)  013422  004322                        .WORD   CRTIM
(5)  013424  010002                        .WORD   ERR6
3764
3765  013426  000402                        BR      3$              ;EXIT
3766
3767  013430  004537  013062          2$:   JSR     R5,AFTER        ;GET REGISTERS
3768  013434  012601                3$:    MOV     (SP)+,R1
3769  013436  000205                        RTS     R5              ;EXIT
3770
3771
3772
3773  013440                                ENDMOD
3774
3775
3776
3777                                 .SBTTL  **TEST 1** - RLCS ADDRESSABILITY
3778
3779  013440                         BGNTST                          ;****START OF TEST****
3780  013440                         STARS
(2)                                  ;;*****************************************************************
3781                                 ;TEST TO SEE IF WE CAN ADDRESS THE CONTROL
3782                                 ;AND STATUS REGISTER.  IF WE TRAP WE WILL REPORT
3783                                 ;THE ERROR AND ABORT.  AFTER THIS TEST WE ONLY KNOW
3784                                 ;THAT WE CAN ADDRESS THE REGISTER.
3785  013440                         STARS
(2)                                  ;;*****************************************************************
3786
3787
3788  013440  005037  002170         1$:    CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
3789  013444                         2$:    SETVEC  ERRVEC,#TRPHAN,#340  ;SET TO CATCH TRAP
(7)  013444  012746  000340                MOV     #340,-(SP)
(6)  013450  012746  013302                MOV     #TRPHAN,-(SP)
(5)  013454  013746  002202                MOV     ERRVEC,-(SP)
```

```
C1TERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 84-15
D2RLAA.P11    05-OCT-77 10:41          **TEST 1** - RLCS ADDRESSABILITY                SEQ 0047

    (4)  013460  012746  000003                MOV     #3,-(SP)
    (3)  013464  104037                        EMT     C$$VEC
    (2)  013466  062706  000010                ADD     #10,SP
3790
3791  013472  005777  166422                   TST     @RLCS           ;ADDRESS RLCS
3792  013476                                   CLRVEC  ERRVEC          ;RELEASE TRAP VECTOR
    (3)  013476  013700  002202                MOV     ERRVEC,R0
    (3)  013502  104036                        EMT     C$CVEC
3793  013504  005737  002170                   TST     TRPFLG          ;TRAP OCCURRED???
3794  013510  001407                           BEQ     3$              ;NO, IKAY PROCEED
3795  013512  013737  002120  002224           MOV     RLCS,GDDAT      ;SET UP ERROR DATA
3796
3797  013520                                   ERRSF   0.,EM1,ERR1     ;BUS TIMEOUT IN ADDRESSING RLCS
    (3)  013520  104461                        TRAP    T$ERCODE
    (5)  013522  000000                        .WORD   0
    (5)  013524  004375                        .WORD   EM1
    (5)  013526  007574                        .WORD   ERR1
3798  013530                           3$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
    (3)  013530  104006                        EMT     C$CLP1
3799  013532                           ENDTST                          ;****END OF TEST****
    (3)  013532                           L10017:
    (3)  013532  104001                        EMT     C$ETST
3800
3801
3802                                     .SBTTL  **TEST 2** - PLBA ADDRESSABILITY
3803
3804  013534                             BGNTST                        ;****START OF TEST****
3805
3806
3807  013534                             STARS
    (2)                                  ;;*****************************************************************
3808                                     ;TEST TO SEE IF WE CAN ADDRESS THE BUS ADDRESS
3809                                     ;REGISTER.  IF WE TRAP WE WILL REPORT THE ERROR
3810                                     ;AND ABORT.  AFTER THIS TEST WE ONLY KNOW THAT
3811                                     ;WE CAN ADDRESS THE REGISTER.
3812  013534                             STARS
    (2)                                  ;;*****************************************************************
3813
3814  013534  005037  002170             1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
3815  013540                             2$:     SETVEC  ERRVEC,#TRPHAN,#340  ;SET TO CATCH TRAP
    (7)  013540  012746  000340                MOV     #340,-(SP)
    (6)  013544  012746  013302                MOV     #TRPHAN,-(SP)
    (5)  013550  013746  002202                MOV     ERRVEC,-(SP)
    (4)  013554  012746  000003                MOV     #3,-(SP)
    (3)  013560  104037                        EMT     C$$VEC
    (2)  013562  062706  000010                ADD     #10,SP
3816
3817  013566  005777  166330                   TST     @RLBA           ;ADDRESS RLBA
3818  013572                                   CLRVEC  ERRVEC          ;RELEASE TRAP VECTOR
    (3)  013572  013700  002202                MOV     ERRVEC,R0
    (3)  013576  104036                        EMT     C$CVEC
3819  013600  005737  002170                   TST     TRPFLG          ;TRAP OCCURRED???
3820  013604  001407                           BEQ     3$              ;NO, CONTINUE
3821  013606  013737  002122  002224           MOV     RLBA,GDDAT      ;SETUP ERROR DATA
3822
3823  013614                                   ERRSF   1.,EM2,ERR1     ;BUS TIMEOUT IN ADDRESSING RLBA
```

```
                (3)  013614  104461                    TRAP    T$ERCODE
                (5)  013616  000001                    .WORD   1
                (5)  013620  004422                    .WORD   EM2
                (5)  013622  007574                    .WORD   ERR1
          3824  013624                         3$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                (3)  013624  104006                    EMT     C$CLP1
          3825  013626                         ENDTST                          ;****END OF TEST****
                (3)  013626                    L10020:
                (3)  013626  104001                    EMT     C$ETST
          3826
          3827
          3828                                .SBTTL  **TEST 3** - RLDA ADDRESSABILITY
          3829
          3830  013630                         BGNTST                          ;****START OF TEST****
          3831  013630                         STARS
                (2)                            ;;**************************************************************
          3832                                 ;TEST TO SEE IF WE CAN ADDRESS THE DISK ADDRESS
          3833                                 ;REGISTER IF WE TRAP WE WILL REPORT THE ERROR
          3834                                 ;AND ABORT.  AFTER THIS TEST WE ONLY KNOW THAT
          3835                                 ;WE CAN ADDRESS THE REGISTER.
          3836  013630                         STARS
                (2)                            ;;**************************************************************
          3837
          3838
          3839  013630  005037  002170         1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
          3840  013634                         2$:     SETVEC  ERRVEC,#TRPHAN,#340  ;SET TO CATCH TRAP
                (7)  013634  012746  000340              MOV     #340,-(SP)
                (6)  013640  012746  013302              MOV     #TRPHAN,-(SP)
                (5)  013644  013746  002202              MOV     ERRVEC,-(SP)
                (4)  013650  012746  000003              MOV     #3,-(SP)
                (3)  013654  104037                      EMT     C$SVEC
                (2)  013656  062706  000010              ADD     #10,SP
          3841
          3842  013662  005777  166236                   TST     @RLDA           ;ADDRESS RLDA
          3843  013666                                   CLRVEC  ERRVEC          ;RELEASE TRAP VECTOR
                (3)  013666  013700  002202              MOV     ERRVEC,R0
                (3)  013672  104036                      EMT     C$CVEC
          3844  013674  005737  002170                   TST     TRPFLG          ;TRAP OCCURRED???
          3845  013700  001407                           BEQ     3$              ;NO, CONTINUE
          3846
          3847  013702  013737  002124  002224         MOV     RLDA,GDDAT      ;SETUP ERROR INFO
          3848  013710                                 ERRSF   2.,EM3,ERR1     ;BUS TIMEOUT IN ADDRESSING RLDA
                (3)  013710  104461                      TRAP    T$ERCODE
                (5)  013712  000002                      .WORD   2
                (5)  013714  004447                      .WORD   EM3
                (5)  013716  007574                      .WORD   ERR1
          3849  013720                         3$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                (3)  013720  104006                      EMT     C$CLP1
          3850  013722                         ENDTST                          ;****END OF TEST****
                (3)  013722                    L10021:
                (3)  013722  104001                      EMT     C$ETST
          3851
          3852
          3853                                .SBTTL  **TEST 4** - RLMP ADDRESSABILITY
          3854
          3855  013724                         BGNTST                          ;****START OF TEST****
```

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

013724  005037  002170          1$:     CLR     TRPFLG              ;CLEAR TRAP OCCURANCE
013730                          2$:     SETVEC  ERRVEC,#TRPHAN,#340  ;SET UP TO CATCH TRAP
013730  012746  000340                  MOV     #340,-(SP)
013734  012746  013302                  MOV     #TRPHAN,-(SP)
013740  013746  002202                  MOV     ERRVEC,-(SP)
013744  012746  000003                  MOV     #3,-(SP)
013750  104037                          EMT     C$SVEC
013752  062706  000010                  ADD     #10,SP

013756  005777  166144                  TST     @RLMP               ;ADDRESS RLMP
013762                                  CLRVEC  ERRVEC              ;RELEASE TRAP VECTOR
013762  013700  002202                  MOV     ERRVEC,R0
013766  104036                          EMT     C$CVEC
013770  005737  002170                  TST     TRPFLG              ;TRAP OCCURRED???
013774  001407                          BEQ     3$                  ;NO, CONTINUE
013776  013737  002126  002224          MOV     RLMP,GDDAT          ;SET UP ERROR INFO

014004                                  ERRSF   3.,EM4,ERR1         ;BUS TIMEOUT IN ADDRESSING RLMP
014004  104461                          TRAP    T$ERCODE
014006  000003                          .WORD   3
014010  004474                          .WORD   EM4
014012  007574                          .WORD   ERR1
014014                          3$:     CKLOOP                      ;CHECK IF /FL:LOE IS SET
014014  104006                          EMT     C$CLP1
014016                          ENDTST                              ;****END OF TEST****
014016                          L10022:
014016  104001                          EMT     C$ETST


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

014020                                  BGNTST                      ;****START OF TEST****


                                        STARS
                                        ;;*****************************************************************
                                        ;TEST THAT WE CAN WRITE/READ BITS 8,9 AND BITS 6-1
                                        ;OF THE CONTROL AND STATUS REGISTER.  BITS 15-10 AND 0
                                        ;ARE DON'T CARE BITS AT THIS TIME AND BIT 7
                                        ;(CONTROLLER READY) IS ALWAYS WRITTEN TO A ONE.
014020                                  STARS
                                        ;;*****************************************************************


014020  012703  002570                  MOV     #CSPAT,R3           ;SET UP TABLE POINTER OF PATTERNS
```

```
3893
3894  014024                          BGNSEG                          ;****START OF SEGMENT****
 (3)  014024  104004                          EMT     C$BSEG
3895
3896  014026                          CSTEST:
3897  014026  011337  002224                 MOV     (R3),GDDAT      ;GET PATTERN INTO GDDAT
3898  014032  052737  000200  002224         BIS     #200,GDDAT      ;INSURE GO IS SET
3899  014040  013777  002224  166052         MOV     GDDAT,@RLCS     ;LOAD RLCS (CONTROL AND STATUS)
3900  014046  032777  040000  166044         BIT     #DERR,@RLCS     ;IF DRIVE ERROR PRESENT
3901  014054  001403                         BEQ     99$             ;THEN EXPECT DRIVE AND
3902  014056  052737  140000  002224         BIS     #ERR!DERR,GDDAT ;COMPOSITE ERROR
3903  014064  017737  166030  002226  99$:    MOV     @RLCS,BDDAT     ;READ RLCS BACK
3904  014072  042737  000001  002226         BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
3905  014100  023737  002224  002226         CMP     GDDAT,BDDAT     ;DID WE READ WHAT WE LOADED
3906  014106  001404                         BEQ     1$              ;YES, THEN BRANCH
3907
3908  014110                                 ERRDF   4.,EM5,ERR2     ;WRONG DATA IN RLCS
 (3)  014110  104462                         TRAP    T$ERCODE
 (5)  014112  000004                         .WORD   4
 (5)  014114  004521                         .WORD   EM5
 (5)  014116  007606                         .WORD   E~~2
3909  014120                          1$:    ESCAPE  SEQ             ;IF /FL:LOE SET LOOP, ELSE EXIT SEQ
 (3)  014120  104010                         EMT     C$ESCAPE
 (3)  014122  000012                         .WORD   10000$-.
3910
3911
3912  014124  005723                         TST     (R3)+           ;BUMP FOR NEXT PATTERN
3913  014126  020327  002666                 CMP     R3,#CSEND       ;CHECK FOR END
3914  014132  001335                         BNE     CSTEST          ;NOT END, LOAD NEXT PATTERN
3915
3916  014134                          ENDSEG                          ;****END OF SEGMENT****
 (3)  014134                          10000$:
 (3)  014134  104005                         EMT     C$ESEG
3917  014136                          ENDTST                          ;****END OF TEST****
 (3)  014136                          L10023:
 (3)  014136  104001                         EMT     C$ETST
3918
3919
3920                                 .SBTTL  **TEST 6** - READ WRITE OF RLBA
3921
3922  014140                         BGNTST                          ;****START OF TEST****
3923
3924  014140                         STARS
 (2)                                 ;;*****************************************************************
3925                                 ;TEST THAT WE CAN WRITE/READ BITS IS THRU 1 OF THE
3926                                 ;BUS ADDRESS REGISTER.  FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
3927                                 ;GROWING 0 AND SHIFTING 0.  BIT 0 IS ALSO LOADED BUT
3928                                 ;SHOULD ALWAYS COME BACK AS 0
3929  014140                         STARS
 (2)                                 ;;*****************************************************************
3930
3931
3932  014140  012703  002256                 MOV     #BEGPAT,R3      ;GET START OF PATTERN LIST
3933  014144                         BGNSEG                          ;****START OF SEGMENT****
 (3)  014144  104004                         EMT     C$BSEG
3934  014146                         BATEST:
```

```
 3935  014146  011337  002224                    MOV     (R3),GDDAT      ;GET PATTERN TO SEND
 3936  014152  005737  002252                    TST     T.CNTLR         ;RL11??
 3937  014156  001403                            BEQ     2$              ;NO,
 3938  014160  042737  000001  002224            BIC     #BIT0,GDDAT     ;KEEP RLBA EVEN (UNIBUS)
 3939  014166  013777  002224  165726  2$:       MOV     GDDAT,@RLBA     ;LOAD PATTERN TO BUS ADDRESS
 3940  014174  017737  165722  002226            MOV     @RLBA,BDDAT     ;READ IT BACK
 3941  014202  023737  002224  002226            CMP     GDDAT,BDDAT     ;IS IT CORRECT?
 3942  014210  001404                            BEQ     1$              ;IF SO, BRANCH
 3943
 3944  014212                                    ERRDF   5.,EM6,ERR2     ;DATA WRONG IN RLBA
  (3)  014212  104462                            TRAP    T$ERCODE
  (5)  014214  000005                            .WORD   5
  (5)  014216  004572                            .WORD   EM6
  (5)  014220  007606                            .WORD   ERR2
 3945  014222                          1$:       ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
  (3)  014222  104010                            EMT     C$ESCAPE
  (3)  014224  000012                            .WORD   10000$-.
```

```
                                                TST     (R3)+           ;BUMP FOR NEXT PATTERN
                                                CMP     R3,#ENDPAT      ;CHECK FOR END
                                                BNE     BA1EST          ;NOT END, BRANCH FOR NEXT

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


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

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

                                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
                                STARS
                                ;;**************************************************************

                                                MOV     #BEGPAT,R3      ;SET UP POINTER TO PATTERN LIST
                                BGNSEG                                  ;****START OF SEGMENT****
                                                EMT     C$BSEG
                                DATEST:
                                                MOV     (R3),GDDAT      ;GET PATTERN
                                                MOV     GDDAT,@RLDA     ;LOAD PATTERN IN DA

                                                MOV     @RLDA,BDDAT     ;READ PATTERN BACK
                                                CMP     GDDAT,BDDAT     ;IS IT CORRECT?
                                                BEQ     1$              ;BRANCH IF CORRECT

                                                ERRDF   6.,EM7,ERR2     ;WRONG DATA IN RLDA
                                                TRAP    T$ERCODE
                                                .WORD   6
                                                .WORD   EM7
                                                .WORD   ERR2
                                1$:             ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
                                                EMT     C$ESCAPE
                                                .WORD   10000$-.


                                                TST     (R3)+           ;BUMP POINTER
                                                CMP     R3,#ENDPAT      ;AT END OF PATTERNS?
                                                BNE     DA1EST          ;NO, BRANCH BACK

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

```
OUTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 95-1
DERLAA.P11     05-OCT-77 10:41          **TEST 7** - READ WRITE OF RLDA                                SEQ 0053

   (3)  014326                          L10025:
   (3)  014326  104001                          EMT     C$ETST
  3988
  3989
  3990                                  .SBTTL  **TEST 8** - BIS OF RLCS
  3991
  3992  014330                          BGNTST                          ;****START OF TEST****
  3993  014330                          STARS
   (2)                                  ;;**************************************************************
  3994                                  ;TEST THAT WE CAN USE THE "BIS" INSTRUCTION ON THE CONTROL
  3995                                  ;AND STATUS REGISTER.  BITS 8,9 AND 6-1 ARE TESTED TO
  3996                                  ;SET INDIVIDUALLY AS WELL AS COLLECTIVELY WITHOUT DESTROYING
  3997                                  ;ANY PREVIOUS DATA PATTERN
  3998  014330                          STARS
   (2)                                  ;;**************************************************************
  3999
  4000
  4001  014330  012703  002570                  MOV     #CSPAT,R3       ;GET BEGINNING OF LIST
  4002  014334                          BGNSEG                          ;****START OF SEGMENT****
   (3)  014334  104004                          EMT     C$BSEG
  4003  014336                          1$:
  4004  014336  012777  000200  165554          MOV     #CRDY,@RLCS     ;INSURE GO IS THERE
  4005  014344  011337  002224                  MOV     (R3),GDDAT      ;SET UP EXPECTED RLCS
  4006  014350  052737  000200  002224          BIS     #CRDY,GDDAT     ;IN GDDAT
  4007  014356  051377  165536                  BIS     (R3),@RLCS      ;BIT SET PATTERN IN RLCS
  4008  014362  032777  040000  165530          BIT     #DERR,@RLCS     ;IF ERROR BIT SET THEN
  4009  014370  001403                          BEQ     99$             ;EXPECT IT ON THE READ
  4010  014372  052737  140000  002224          BIS     #ERR!DERR,GDDAT ;BACK
  4011  014400  017737  165514  002226  99$:    MOV     @RLCS,BDDAT     ;READ RLCS TO CHECK "BIS"
  4012  014406  042737  000001  002226          BIC     #DRDY,BDDAT     ;CLEAR OUT DRIVE READY
  4013  014414  023737  002226  002224          CMP     BDDAT,GDDAT     ;DID BIS WORK?
  4014  014422  001404                          BEQ     2$              ;BRANCH IF OKAY
  4015
  4016  014424                                  ERRDF   7.,EM61,ERR2    ;WRONG DATA IN RLCS
   (3)  014424  104462                          TRAP    T$ERCODE
   (5)  014426  000007                          .WORD   7
   (5)  014430  006255                          .WORD   EM61
   (5)  014432  007606                          .WORD   ERR2
  4017  014434                          2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
   (3)  014434  104010                          EMT     C$ESCAPE
   (3)  014436  000012                          .WORD   10000$-.
  4018                                                                  ;BIT OR CLEARED OTHER BIT
  4019
  4020  014440  005723                          TST     (R3)+           ;GET NEXT PATTERN
  4021  014442  022703  002666                  CMP     #CSEND,R3       ;AT END OF LIST
  4022  014446  001333                          BNE     1$              ;NO GO BACK FOR TEST OF
  4023                                                                  ;NEXT PATTERN
  4024  014450                          ENDSEG                          ;****END OF SEGMENT****
   (3)  014450                          10000$:
   (3)  014450  104005                          EMT     C$ESEG
  4025  014452                          ENDTST                          ;****END OF TEST****
   (2)  014452                          L10026:
   (3)  014452  104001                          EMT     C$ETST
  4026
  4027
  4028                                  .SBTTL  **TEST 9** - BIC OF RLCS
```

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

                                STARS
                                ;;*****************************************************************
                                ;TEST THAT THE "BIC" INSTRUCTION WILL WORK ON THE
                                ;CONTROL AND STATUS REGISTER.  BITS 8-9 AND 6-1 ARE
                                ;TESTED.
                                STARS
                                ;;*****************************************************************

014454 012703 002570                    MOV     #CSPAT,R3       ;GET BEGINNING OF PATTERNS
014460                          BGNSEG                          ;****START OF SEGMENT****
014460 104004                           EMT     C$BSEG
014462                          1$:
014462 012777 001776 165430             MOV     #1776,@RLCS     ;SET ALL SETTABLE BITS
014470 012737 001776 002224             MOV     #1776,GDDAT     ;SET UP EXPECT DATA IN
014476 041337 002224                    BIC     (R3),GDDAT      ;GDDAT
014502 041377 165412                    BIC     (R3),@RLCS      ;CLEAR BITS IN RLCS VIA "BIC"
014506 032777 040000 165404             BIT     #DERR,@RLCS     ;IF DRIVE ERROR BIT SET
014514 001403                           BEQ     99$             ;EXPECT IT SET WHEN WE
014516 052737 140000 002224             BIS     #ERR!DERR,GDDAT ;READ IT BACK
014524 017737 165370 002226     99$:    MOV     @RLCS,BDDAT     ;MOVE RLCS TO BDDAT FOR COMPARE
014532 042737 000001 002226             BIC     #DRDY,BDDAT     ;CLEAR DRIVE READY
014540 023737 002226 002224             CMP     BDDAT,GDDAT     ;DID "BIC" WORK PROPERLY
014546 001404                           BEQ     2$              ;BRANCH IF OKAY

014550                                  ERRDF - 8.,EM62,ERR2    ;WRONG DATA IN RLCS
014550 104462                           TRAP    T$ERCODE
014552 000010                           .WORD   8
014554 006336                           .WORD   EM62
014556 007606                           .WORD   ERR2
014560                          2$:     ESCAPE  SEQ             ;IF /FL:LOE SET LOOP, ELSE EXIT SEQ
014560 104010                           EMT     C$ESCAPE
014562 000012                           .WORD   10000$-.

014564 005723                           TST     (R3)+           ;GET NEXT PATTERN
014566 020327 002666                    CMP     R3,#CSEND       ;AT END OF LIST
014572 001333                           BNE     1$              ;NO, GO BACK WITH NEXT PATTERN
014574                          ENDSEG                          ;****END OF SEGMENT****
014574                          10000$:
014574 104005                           EMT     C$ESEG
014576                          ENDTST                          ;****END OF TEST****
014576                          L10027:
014576 104001                           EMT     C$ETST


                                .SBTTL  **TEST 10** - BIS OF RLBA

014600                          BGNTST                          ;****START OF TEST****

014600                          STARS
                                ;;*****************************************************************
                                ;TEST THAT THE "BIS" INSTRUCTION WILL WORK ON THE BUS
                                ;ADDRESS REGISTER.  BITS 15-0 ARE LOADED, ONLY BITS 15-1
```

```
                                        ;ARE EXPECTED BACK.  FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
                                        ;GROWING 0, AND SHIFTING 0.
        014600                          STARS
                                        ;;*****************************************************************


        014600  012703  002256                  MOV     #BEGPAT,R3      ;GET START OF LIST
        014604                          BGNSEG                          ;****START OF SEGMENT****
        014604  104004                          EMT     C$BSEG
        014606                          1$:
        014606  005077  165310                  CLR     @RLBA           ;CLEAR "BA"
        014612  011337  002224                  MOV     (R3),GDDAT      ;SET EXPECTED
        014616  005737  002252                  TST     T.CNTLR         ;RL11
        014622  001403                          BEQ     3$              ;NO
        014624  042737  000001  002224          BIC     #1,GDDAT        ;BIT 0 CAN'T SET IN RLBA (UNIBLS
        014632  051377  165264          3$:     BIS     (R3),@RLBA      ;BIS RLBA WITH PATTERN
        014636  017737  165260  002226          MOV     @RLBA,BDDAT     ;READ "BA"
        014644  023737  002226  002224          CMP     BDDAT,GDDAT     ;DID RLBA LOAD PROPERLY?
        014652  001404                          BEQ     2$              ;BRANCH IF YES

        014654                                  ERRDF   9.,EM63,ERR2    ;WRONG DATA IN RLBA
        014654  104462                          TRAP    T$ERCODE
        014656  000011                          .WORD   9
        014660  006421                          .WORD   EM63
        014662  007606                          .WORD   ERR2
        014664                          2$:     ESCAPE  SEG             ;IF  FL:.OE SET LOOP, ELSE EXIT SEG
        014664  104010                          EMT     C$ESCAPE
        014666  000012                          .WORD   10000$-.

        014670  005723                          TST     (R3)+           ;GET NEXT PATTERN
        014672  020327  002464                  CMP     R3,#ENDPAT      ;DID WE COMPLETE LIST
        014676  001343                          BNE     1$              ;NO, GO BACK FOR NEXT.
        014700                          ENDSEG                          ;****END OF SEGMENT****
        014700                          10000$:
        014700  104005                          EMT     C$ESEG
        014702                          ENDTST                          ;****END OF TEST****
        014702                          L10030:
        014702  104001                          EMT     C$ETST



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

        014704                          BGNTST                          ;****START OF TEST****

        014704                          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.
        014704                          STARS
                                        ;;*****************************************************************


        014704  012703  002256                  MOV     #BEGPAT,R3      ;GET START OF LIST
        014710                          BGNSEG                          ;****START OF SEGMENT****
        014710  104004                          EMT     C$BSEG
```

```
OUTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 85-4
DZRLAA.P11    05-OCT-77 10:41          **TEST 11** - BIC OF RLBA                                    SEQ 0056

 4112  014712                    1$:
 4113  014712  012777  177776  165202          MOV     #-2,@RLBA       ;SET RLBA TO ALL 1'S (BIT 0=0)
 4114  014720  012737  177776  002224          MOV     #-2,GDDAT       ;SET UP EXPECTED RESULTS
 4115  014726  041337  002224                  BIC     (R3),GDDAT      ;IN GDDAT
 4116  014732  041377  165164                  BIC     (R3),@RLBA      ;BIC RLBA
 4117  014736  017737  165160  002226          MOV     @RLBA,BDDAT     ;READ RLBA
 4118  014744  023737  002226  002224          CMP     BDDAT,GDDAT     ;BIC WORK OKAY?
 4119  014752  001404                          BEQ     2$              ;IF YES BRANCH
 4120
 4121  014754                                  ERRDF   10.,EM64,ERR2   ;WRONG DATA IN RLBA
  (3)  014754  104462                          TRAP    T$ERCODE
  (5)  014756  000012                          .WORD   10
  (5)  014760  006502                          .WORD   EM64
  (5)  014762  007606                          .WORD   ERR2
 4122  014764                    2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
  (3)  014764  104010                          EMT     C$ESCAPE
  (3)  014766  000012                          .WORD   10000$-.
 4123
 4124  014770  005723                          TST     (R3)+           ;GET NEXT PATTERN
 4125  014772  020327  002464                  CMP     R3,#ENDPAT      ;HAVE WE COMPLETED LIST
 4126  014776  001345                          BNE     1$              ;NO, GO BACK FOR NEXT
 4127  015000                    ENDSEG                                ;****END OF SEGMENT****
  (3)  015000                    10000$:
  (3)  015000  104005                          EMT     C$ESEG
 4128  015002                    ENDTST                                ;****END OF TEST****
  (3)  015002                    L10031:
  (3)  015002  104001                          EMT     C$ETST
 4129
 4130
 4131                            .SBTTL  **TEST 12** - BIS OF RLDA
 4132
 4133  015004                    BGNTST                                ;****START OF TEST****
 4134
 4135  015004                    STARS
  (2)                            ;;*****************************************************************
 4136                            ;TEST THAT THE "BIS" INSTRUCTION WILL WORK ON THE DISK ADDRESS
 4137                            ;REGISTER.  BITS 15-0 ARE TESTED WITH 4 PATTERNS, GROWING 1,
 4138                            ;SHIFTING 1, GROWING 0, AND SHIFTING 0.
 4139  015004                    STARS
  (2)                            ;;*****************************************************************
 4140
 4141
 4142  015004  012703  002256                  MOV     #BEGPAT,R3      ;GET START OF LIST
 4143  015010                    BGNSEG                                ;****START OF SEGMENT****
  (3)  015010  104004                          EMT     C$BSEG
 4144  015012                    1$:
 4145  015012  005077  165106                  CLR     @RLDA           ;CLEAR "DA"
 4146  015016  011337  002224                  MOV     (R3),GDDAT      ;SET EXPECTED
 4147  015022  051377  165076                  BIS     (R3),@RLDA      ;BIS RLDA
 4148  015026  017737  165072  002226          MOV     @RLDA,BDDAT     ;READ RLDA
 4149  015034  023737  002226  002224          CMP     BDDAT,GDDAT     ;IS RLDA CORRECT
 4150  015042  001404                          BEQ     2$              ;IF OKAY BRANCH
 4151
 4152  015044                                  ERRDF   11.,EM65,ERR2   ;WRONG DATA IN RLDA
  (3)  015044  104462                          TRAP    T$ERCODE
  (5)  015046  000013                          .WORD   11
```

```
Q.TERR *MACY11 30(1046)  30-OCT-77  16:51  PAGE 85-5
DZRLAA.P11    05-OCT-77 10:41        **TEST 12** - BIS OF RLDA                                SEQ 0057

  (5)  015050  006565                          .WORD   EM65
  (5)  015052  007606                          .WORD   ERR2
 4153  015054                       2$:        ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
  (3)  015054  104010                          EMT     C$ESCAPE
  (3)  015056  000012                          .WORD   10000$-.
 4154
 4155  015060  005723                          TST     (R3)+           ;GET NEXT PATTERN
 4156  015062  020327  002464                  CMP     R3,#ENDPAT      ;HAVE WE FINISHED?
 4157  015066  001351                          BNE     1$              ;NO GO BACK
 4158  015070                       ENDSEG                             ;****END OF SEGMENT****
  (3)  015070                       10000$:
  (3)  015070  104005                          EMT     C$ESEG
 4159  015072                       ENDTST                             ;****END OF TEST****
  (3)  015072                       L10032:
  (3)  015072  104001                          EMT     C$ETST
 4160
 4161
 4162                              .SBTTL  **TEST 13** - BIC OF RLDA
 4163
 4164  015074                       BGNTST                             ;****START OF TEST****
 4165
 4166  015074                       STARS
  (2)                              ;;*****************************************************************
 4167                              ;TEST THAT THE "BIC" INSTRUCTION WORKS ON THE DISK
 4168                              ;ADDRESS REGISTER.  ALL BITS ARE TESTED WITH FOUR
 4169                              ;PATTERNS: GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
 4170  015074                       STARS
  (2)                              ;;*****************************************************************
 4171
 4172
 4173  015074  012703  002256                  MOV     #BEGPAT,R3      ;GET START OF LIST
 4174  015100                       BGNSEG                             ;****START OF SEGMENT****
  (3)  015100  104004                          EMT     C$BSEG
 4175  015102                       1$:
 4176  015102  012777  177777  165014          MOV     #-1,@RLDA       ;SET RLDA TO ALL 1'S
 4177  015110  012737  177777  002224          MOV     #-1,GDDAT       ;SET EXPECTED DATA
 4178  015116  041337  002224                  BIC     (R3),GDDAT      ;SET EXPECTED DATA
 4179  015122  041377  164776                  BIC     (R3),@RLDA      ;"BIC" RLDA
 4180  015126  017737  164772  002226          MOV     @RLDA,BDDAT     ;READ RLDA
 4181  015134  023737  002224  002226          CMP     GDDAT,BDDAT     ;DID "BIC" WORK?
 4182  015142  001404                          BEQ     2$              ;IF IT DID BRANCH
 4183
 4184  015144                               ERRDF   12.,EM66,ERR2      ;WRONG DATA IN RLDA
  (3)  015144  104462                          TRAP    T$ERCODE
  (5)  015146  000014                          .WORD   12
  (5)  015150  006646                          .WORD   EM66
  (5)  015152  007606                          .WORD   ERR2
 4185  015154                       2$:        ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
  (3)  015154  104010                          EMT     C$ESCAPE
  (3)  015156  000012                          .WORD   10000$-.
 4186
 4187  015160  005723                          TST     (R3)+           ;GET NEXT PATTERN
 4188  015162  020327  002464                  CMP     R3,#ENDPAT      ;DONE?
 4189  015166  001345                          BNE     1$              ;NO GO BACK
 4190  015170                       ENDSEG                             ;****END OF SEGMENT****
  (3)  015170                       10000$:
```

```
(3)  015170  104005                        EMT     C$ESEG
4191 015172                         ENDTST                          ;****END OF TEST****
(3)  015172                         L10033:
(3)  015172  104001                        EMT     C$ETST
4192
4193
4194                                .SBTTL  **TEST 14** - BUS RESET OF RLCS
4195
4196 015174                         BGNTST                          ;****START OF TEST****
4197
4198 015174                         STARS
(2)                                 ;;*****************************************************************
4199                                ;TEST THAT A BUS RESET WILL CLEAR THE PROPER BITS
4200                                ;OF THE CONTROL AND STATUS REGISTER.  THOSE BITS ARE
4201                                ;BITS 6-1,8,9,10,11,12,13,15.  BIT 15 WILL CLEAR ONLY
4202                                ;IF BIT 14 (DRIVE ERROR IS NOT SET).  BIT 0 (DRIVE READY)
4203                                ;IS A DON'T CARE.  IF AT THE START UP THIS TEST BIT
4204                                ;14 (DRIVE ERROR) IS SET WE WILL INSIST IF IS THERE AFTER
4205                                ;THE "RESET" ALONG WITH BIT 15  COMPOSITE ERROR).  BITS
4206                                ;15-10 ARE NOT WRITEABLE.
4207 015174                         STARS
(2)                                 ;;*****************************************************************
4208
4209
4210 015174                                 SETPRI  #PRI07          ;PRIORITY TO SEVEN
(3)  015174  012700  000340                MOV     #PRI07,R0
(3)  015200  104041                        EMT     C$SPRI
4211 015202  012777  000377  164710        MOV     #377,@RLCS      ;LOAD ALL RLCS LOADABLE BITS
4212 015210  012737  000200  002224        MOV     #CRDY,GDDAT     ;SETUP EXPECTED
4213 015216  032777  040000  164674        BIT     #DERR,@RLCS     ;DRIVE ERR SET?
4214 015224  001403                        BEQ     1$              ;IF NOT DON'T EXPECT IT
4215 015226  052737  140000  002224        BIS     #DERR!ERR,GDDAT ;IT'S SET, INIT BETTER NOT CLR
4216 015234  012700  000100         1$:    MOV     #100,R0         ;SET UP A WAIT LOOP
4217 015240                                BRESET                  ;BUS RESET
(3)  015240  104033                        EMT     C$RESET
4218 015242  005300                 2$:    DEC     R0              ;WAIT IN CASE OF DRIVE ERROR
4219 015244  001376                        BNE     2$              ;
4220 015246  017737  164646  002226        MOV     @RLCS,BDDAT     ;READ RLCS
4221 015254  042737  000001  002226        BIC     #DRDY,BDDAT     ;CLEAR OUT DRDY - DON'T CARE
4222 015262  023737  002226  002224        CMP     BDDAT,GDDAT     ;DID INIT WORK
4223 015270  001404                        BEQ     3$              ;YES, BRANCH
4224
4225 015272                                ERRDF   13.,EM67,ERR2   ;WRONG DATA IN RLCS
(3)  015272  104462                        TRAP    T$ERCODE
(5)  015274  000015                        .WORD   13
(5)  015276  006731                        .WORD   EM67
(5)  015300  007606                        .WORD   ERR2
4226 015302                         3$:
4227 015302                         ENDTST                          ;****END OF TEST****
(3)  015302                         L10034:
(3)  015302  104001                        EMT     C$ETST
4228
4229
4230                                .SBTTL  **TEST 15** - BUS RESET OF RLBA
4231
4232 015304                         BGNTST                          ;****START OF TEST****
```

```
                                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
                                STARS
                                ;;*****************************************************************

015304  012777  177776  164610          MOV     #-2,@RLBA       ;SET BA TO ALL 1'S
015312  005737  002252                  TST     T.CNTLR         ;RL11??
015316  001403                          BEQ     2$              ;NO
015320  052777  000001  164574          BIS     #1,@RLBA
015326  005037  002224          2$:     CLR     GDDAT           ;CLEAR EXPECTED DATA
015332                                  BRESET                  ;ISSUE BUS INIT
015332  104033                          EMT     C$RESET
015334  017737  164562  002226          MOV     @RLBA,BDDAT     ;READ RLBA
015342  001404                          BEQ     1$              ;IF CLEAR BRANCH

015344                                  ERRDF   14.,EM70.ERR2   ;WRONG DATA IN RLBA
015344  104462                          TRAP    T$ERCODE
015346  000016                          .WORD   14
015350  006766                          .WORD   EM70
015352  007606                          .WORD   ERR2
015354                          1$:

015354                          ENDTST                          ;****END OF TEST****
015354                          L10035:
015354  104001                          EMT     C$ETST


                                .SBTTL  **TEST 16** - BUS RESET OF RLDA

015356                          BGNTST                          ;****START OF TEST****

015356                          STARS
                                ;;*****************************************************************
                                ;TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
                                ;DISK ADDRESS REGISTER.  THE DISK ADDRESS IS LOADED WITH 177777
                                ;AND IS EXPECTED TO BE ZERO AFTER THE RESET.
015356                          STARS
                                ;;*****************************************************************

015356  012777  177777  164540          MOV     #-1,@RLDA       ;SET DA TO ALL 1'S
015364  005037  002224                  CLR     GDDAT           ;CLEAR EXPECTED
015370                                  BRESET                  ;ISSUE BUS INIT
015370  104033                          EMT     C$RESET
015372  017737  164526  002226          MOV     @RLDA,BDDAT     ;READ RLDA
015400  001404                          BEQ     1$              ;IF CLEAR BRANCH

015402                                  ERRDF   15.,EM71.ERR2   ;WRONG DATA IN RLDA
015402  104462                          TRAP    T$ERCODE
015404  000017                          .WORD   15
015406  007023                          .WORD   EM71
```

```
D2ERR  MACY11 30.1046(  30-OCT-77  16:51  PAGE 85-8
D2RLAA.P11    05-OCT-77 10:41        **TEST 16** - BUS RESET OF RLDA                                SEQ 0086

(5) 015410  007606                          .WORD   ERR2
4274  015412                          1$:
4275
4276  015412                          ENDTST                          ;****END OF TEST****
(3) 015412                          L10036:
(3) 015412  104001                          EMT     C$ETST
4277
4278
4279                                .SBTTL  **TEST 17** - UNIQUENESS OF RLCS
4280
4281  015414                        BGNTST                            ;****START OF TEST****
4282
4283  015414                        STARS
(2)                                 ;;*****************************************************************
4284                                ;TEST THE UNIQUENESS OF THE CONTROL AND STATUS
4285                                ;REGISTER.  THE RLBA AND RLDA ARE PRELOADED WITH
4286                                ;177776 AND 177777 RESPECTIVELY.  THE RLCS IS THEN
4287                                ;LOADED TO INSURE THAT NEITHER THE RLBA OR RLDA
4288                                ;ARE MODIFIED BY THE WRITING OF THE RLCS.
4289  015414                        STARS
(2)                                 ;;*****************************************************************
4290
4291
4292  015414  012737  000201  002174          MOV     #DRDY!CRDY,@LDCSR       ;SET DRIVE AND CONTROLLER READY
4293  015422  012777  177776  164472          MOV     #-2,@RLBA               ;SET RLBA TO ALL 1'S
4294  015430  012777  177777  164466          MOV     #-1,@RLDA               ;SET RLDA TO ALL 1'S
4295  015436  013777  002174  164454          MOV     @LDCSR,@RLCS            ;WRITE RLCS
4296
4297                                ;CHECK THAT RLBA REMAINED UNEFFECTED
4298
4299  015444  022777  177776  164450          CMP     #-2,@RLBA               ;RLBA OKAY?
4300  015452  001412                          BEQ     1$                      ;YES, GO CHECK DA
4301
4302  015454  012737  177776  002224          MOV     #-2,GDDAT               ;SET UP EXPECTED
4303  015462  017737  164434  002226          MOV     @RLBA,BDDAT             ;READ RLBA
4304
4305  015470                                  ERRDF   16.,EM72,ERR2           ;CS MODIFIED BA
      015470  104462                          TRAP    T$ERCODE
      015472  000020                          .WORD   16
      015474  007060                          .WORD   EM72
      015476  007606                          .WORD   ERR2
4306  015500                          1$:     CKLOOP                          ;CHECK IF /FLOOP IS SET
      015500  104006                          EMT     C$CLP1
4307
4308  015502  022777  177777  164414          CMP     #-1,@RLDA               ;RLDA OKAY?
4309  015510  001412                          BEQ     2$                      ;YES, CONTINUE
4310
4311  015512  012737  177777  002224          MOV     #-1,GDDAT               ;SET UP EXPECTED
4312  015520  017737  164400  002226          MOV     @RLDA,BDDAT             ;READ DA
4313
4314  015526                                  ERRDF   17.,EM73,ERR2           ;CS MODIFIED DA
      015526  104462                          TRAP    T$ERCODE
      015530  000021                          .WORD   17
      015532  007113                          .WORD   EM73
      015534  007606                          .WORD   ERR2
4315  015536                          2$:
```

```
                                ENDTST                          ;****END OF TEST****
015536                          L10037:
(3) 015536
(3) 015536  104001                      EMT     C$ETST

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

015540                          BGNTST                          ;****START OF TEST****
015540                          STARS
(2)                             ;;*****************************************************************
                                ;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.
015540                          STARS
(2)                             ;;*****************************************************************

015540  012737  000200  002224          MOV     #CRDY,GDDAT     ;CONTROLLER READY
015546  032777  040000  164344          BIT     #DERR,@RLCS     ;IF DRIVE ERROR IS
015554  001403                        BEQ     99$             ;SET THEN EXPECT IT
015556  052737  140000  002224          BIS     #ERR!DERR,GDDAT ;SET WHEN WE READ IT.
015564  013777  002224  164326  99$:    MOV     GDDAT,@RLCS     ;LOAD RLCS
015572  012777  177777  164324          MOV     #-1,@RLDA       ;LOAD RLDA
015600  005077  164316                  CLR     @RLBA           ;CLEAR RLBA

                                ;CHECK IF RLCS IS OKAY

015604  017737  164310  002226          MOV     @RLCS,BDDAT     ;READ RLCS
015612  042737  000001  002226          BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
015620  023737  002226  002224          CMP     BDDAT,GDDAT     ;CS OK?
015626  001404                        BEQ     1$              ;YES, GO CHECK DA

015630                          ERRDF   18.,EM74,ERR2   ;BA MODIFIED CS
(3) 015630  104462              TRAP    T$ERCODE
(5) 015632  000022              .WORD   18
(5) 015634  007146              .WORD   EM74
(5) 015636  007606              .WORD   ERR2
015640                          1$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
(3) 015640  104006              EMT     C$CLP1

015642  022777  177777  164254          CMP     #-1,@RLDA       ;IS RLDA OKAY?

015650  001412                        BEQ     2$              ;IF OKAY BRANCH

015652  012737  177777  002224          MOV     #-1,GDDAT       ;SET UP EXPECTED
015660  017737  164240  002226          MOV     @RLDA,BDDAT     ;READ RLDA

015666                          ERRDF   19.,EM75,ERR2   ;BA MODIFIED DA
(3) 015666  104462              TRAP    T$ERCODE
(5) 015670  000023              .WORD   19
(5) 015672  007200              .WORD   EM75
(5) 015674  007606              .WORD   ERR2
```

```
                                        2$:
                                        ENDTST                          ;****END OF TEST****
                                        L10040:
 (3)  015676                                    EMT     C$ETST
4361
4362
4363                                    .SBTTL  **TEST 19** - UNIQUENESS OF RLDA
4364
4365  015700                            BGNTST                          ;****START OF TEST****
4366
4367
4368  015700                            STARS
 (2)                                    ;;**************************************************************
4369                                    ;TEST THE UNIQUENESS OF THE DISK ADDRESS REGISTER.  THE RLCS
4370                                    ;AND RLBA ARE LOADED WITH XXX20X AND 177776
4371                                    ;RESPECTIVELY.  THE RLDA IS THEN WRITTEN TO INSURE
4372                                    ;THAT NEITHER THE RLCS OR THE RLBA ARE MODIFIED
4373                                    ;BY WRITING THE RLDA.
4374  015700                            STARS
 (2)                                    ;;**************************************************************
4375
4376
4377  015700  012737  000200  002224            MOV     #CRDY,GDDAT     ;CONTROLLER READY
4378  015706  032777  040000  164204            BIT     #DERR,@RLCS     ;IF DRIVE ERROR SET
4379  015714  001403                            BEQ     99$             ;THEN EXPECT IT LATER
4380  015716  052737  140000  002224            BIS     #ERR!DERR,GDDAT
4381  015724  013777  002224  164166  99$:     MOV     GDDAT,@RLCS     ;LOAD CS
4382  015732  012777  177776  164162            MOV     #-2,@RLBA       ;LOAD BA WITH ALL 1'S
4383  015740  005077  164160                    CLR     @RLDA           ;CLEAR RLDA
4384
4385                                    ;CHECK IF RLCS IS OKAY
4386
4387  015744  017737  164150  002226            MOV     @RLCS,BDDAT     ;READ RLCS
4388  015752  042737  000001  002226            BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
4389  015760  023737  002224  002226            CMP     GDDAT,BDDAT     ;RLCS OKAY?
4390  015766  001404                            BEQ     1$              ;YES, THEN BRANCH
4391
4392  015770                                    ERRDF   20.,EM76,ERR2   ;DA MODIFIED CS
 (3)  015770  104462                            TRAP    T$ERCODE
 (5)  015772  000024                            .WORD   20
 (5)  015774  007232                            .WORD   EM76
 (5)  015776  007606                            .WORD   ERR2
4393  016000                            1$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  016000  104006                            EMT     C$CLP1
4394
4395  016002  022777  177776  164112            CMP     #-2,@RLBA       ;IS RLBA OKAY?
4396  016010  001412                            BEQ     2$              ;BRANCH IF OKAY
4397
4398  016012  012737  177776  002224            MOV     #-2,GDDAT       ;SET UP EXPECTED
4399  016020  017737  164076  002226            MOV     @RLBA,BDDAT     ;READ RLBA
4400
4401  016026                                    ERRDF   21.,EM77,ERR2   ;DA MODIFIED BA
 (3)  016026  104462                            TRAP    T$ERCODE
 (5)  016030  000025                            .WORD   21
 (5)  016032  007265                            .WORD   EM77
 (5)  016034  007606                            .WORD   ERR2
```

```
                                        2$:

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

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

                                        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
                                        ;;*****************************************************************

016040  012737  000200  002224          MOV     #CRDY,GDDAT     ;CONTROLLER READY
016046  032777  040000  164044          BIT     #DERR,@RLCS     ;IF DRIVE ERROR SET
016054  001403                          BEQ     99$             ;THE EXPECT IT LATER
016056  052737  140000  002224          BIS     #ERR!DERR,GDDAT
016064  013777  002224  164026  99$:    MOV     GDDAT,@RLCS     ;LOAD CS
016072  012777  177776  164022          MOV     #-2,@RLBA       ;LOAD BA WITH ALL 1'S
016100  012777  177777  164016          MOV     #-1,@RLDA       ;LOAD RLDA
016106  005077  164014                  CLR     @RLMP           ;WRITE RLMP

                                        ;CHECK IF RLCS IS OKAY

016112  017737  164002  002226          MOV     @RLCS,BDDAT     ;READ RLCS
016120  042737  000001  002226          BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
016126  023737  002224  002226          CMP     GDDAT,BDDAT     ;RLCS OKAY?
016134  001404                          BEQ     1$              ;YES, THEN BRANCH

016136                                  ERRDF   201.,EM44,ERR2  ;MP MODIFIED CS
016136  104462                          TRAP    T$ERCODE
016140  000311                          .WORD   201
016142  005601                          .WORD   EM44
016144  007606                          .WORD   ERR2
016146                          1$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
016146  104006                          EMT     C$CLP1

016150  022777  177776  163744          CMP     #-2,@RLBA       ;IS RLBA OKAY?
016156  001412                          BEQ     2$              ;BRANCH IF OKAY

016160  012737  177776  002224          MOV     #-2,GDDAT       ;SET UP EXPECTED
016166  017737  163730  002226          MOV     @RLBA,BDDAT     ;READ RLBA

016174                                  ERRDF   211.,EM45,ERR2  ;MP MODIFIED BA
016174  104462                          TRAP    T$ERCODE
016176  000323                          .WORD   211
016200  005634                          .WORD   EM45
```

```
                                .WORD   ERR2
4446    016204                  2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
(3)     016204  104006                  EMT     C$CLP1
4447    016206  022777  177777  163710          CMP     #-1,@RLDA       ;DISK ADDRESS OKAY
4448    016214  001412                  BEQ     3$              ;YES, CONTINUE
4449
4450    016216  017737  163702  002226          MOV     @RLDA,BDDAT     ;SET UP BAD
4451    016224  012737  177777  002224          MOV     #-1,GDDAT       ;SET UP EXPECTED
4452
4453    016232                          ERRDF   212.,EM46,ERR2  ;MP MODIFIED DA
(3)     016232  104462                  TRAP    T$ERCODE
(5)     016234  000324                  .WORD   212
(5)     016236  005667                  .WORD   EM46
(5)     016240  007606                  .WORD   ERR2
4454
4455    016242                  3$:
4456
4457
4458    016242                  ENDTST                          ;****END OF TEST****
(3)     016242                  L10042:
(3)     016242  104001                  EMT     C$ETST
4459
4460                            .SBTTL  **TEST 21** - NOOP FUNCTION(RL11 ONLY)
4461
4462    016244                  BGNTST                          ;****START OF TEST****
4463
4464
4465
4466    016244                  STARS
(2)                             ;;*****************************************************************
4467                            ;TEST THAT NOOP WILL FUNCTION.  WE WILL ISSUE THE
4468                            ;NOOP AND WAIT FOR CONTROLLER READY TO SET.  A
4469                            ;TIMEOUT OF 200 MILLISECS IS ALLOWED.  DRIVE 0 IS ALWAYS
4470                            ;SELECTED SINCE THE DRIVE IS NOT NECESSARY.
4471    016244                  STARS
(2)                             ;;*****************************************************************
4472
4473
4474    016244  005737  002252          TST     T.CNTLR         ;RLV11??
4475    016250  001410                  BEQ     99$             ;YES SKIP TEST
4476
4477
4478    016252  004537  012552          JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
4479    016256  000000                  NOOPO                   ;NOOP(0) FUNCTION
4480    016260  004537  013362          JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
4481    016264                  2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
(3)     016264  104006                  EMT     C$CLP1
4482
4483    016266  004537  012252          JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
4484
4485    016272                  99$:
4486    016272                  ENDTST                          ;****END OF TEST****
(3)     016272                  L10043:
(3)     016272  104001                  EMT     C$ETST
4487
4488
```

```
                                .SBTTL  **TEST 22** - TEST NOOP DOES NOTHING

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

                                STARS
                                ;;*****************************************************************
                                ;TEST THAT ISSUING A NOOP FUNCTION DOES NOTHING.  THIS IS DONE BY WRITING
                                ;THE RLBA, AND RLDA, READING THE RLMP AND MAKING SURE NOTHING CHANGES.
                                STARS
                                ;;*****************************************************************

016274  005737  002252                TST     T.CNTLR         ;RLV11??
016300  001476                        BEQ     3$

016302  012777  000001  163614        MOV     #1,@RLDA        ;LOAD DISK ADDRESS
016310  012777  000002  163604        MOV     #2,@RLBA        ;LOAD BUS ADDRESS
016316  005077  163604                CLR     @RLMP
016322  017737  163600  002224        MOV     @RLMP,GDDAT     ;READ RLMP

016330  004537  012552                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
016334  000000                        NOOPO
016336  004537  013362                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
016342                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
(3) 016342  104006                    EMT     C$CLP1

016344  004537  012252                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
016350                                ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 016350  104010                    EMT     C$ESCAPE
(3) 016352  000124                    .WORD   L10044-.

016354  017737  163546  002226        MOV     @RLMP,BDDAT     ;READ RLMP
016362  023737  002224  002226        CMP     GDDAT,BDDAT     ;RLMP OK?
016370  001404                        BEQ     1$

016372                                ERRDF   202.,EM14,ERR2
(3) 016372  104462                    TRAP    T$ERCODE
(5) 016374  000312                    .WORD   202
(5) 016376  004741                    .WORD   EM14
(5) 016400  007606                    .WORD   ERR2

016402                          1$:   CKLOOP                  ;CHECK IF /FL:LOE IS SET
(3) 016402  104006                    EMT     C$CLP1

016404  012737  000002  002224        MOV     #2,GDDAT        ;SET UP EXP'D BA
016412  017737  163504  002226        MOV     @RLBA,BDDAT     ;READ BA
016420  023737  002224  002226        CMP     GDDAT,BDDAT     ;BA OK?
016426  001404                        BEQ     2$              ;YES

016430                                ERRDF   203.,EM15,ERR2
(3) 016430  104462                    TRAP    T$ERCODE
(5) 016432  000313                    .WORD   203
(5) 016434  004767                    .WORD   EM15
(5) 016436  007606                    .WORD   ERR2

016440                          2$:   CKLOOP                  ;CHECK IF /FL:LOE IS SET
(3) 016440  104006                    EMT     C$CLP1
```

```
4530
4531  016442  012737  000001  002224          MOV     #1,GDDAT        ;SET UP EXP'D DA
4532  016450  017737  163450  002226          MOV     @RLDA,BDDAT     ;READ DA
4533  016456  023737  002224  002226          CMP     GDDAT,BDDAT     ;DA OKAY
4534  016464  001404                          BEQ     3$
4535
4536  016466                                  ERRDF   204.,EM16,ERR2
 (3)  016466  104462                          TRAP    T$ERCODE
 (5)  016470  000314                          .WORD   204
 (5)  016472  005015                          .WORD   EM16
 (5)  016474  007606                          .WORD   ERR2
4537
4538  016476                          3$:
4539
4540  016476                          ENDTST                          ;****END OF TEST****
 (3)  016476                          L10044:
 (3)  016476  104001                          EMT     C$ETST
4541
4542
4543                                  .SBTTL  **TEST 23** - TEST OF INTERRUPT
4544
4545  016500                          BGNTST                          ;****START OF TEST****
4546
4547  016500                          STARS
 (2)                                  ;*****************************************************************
4548                                  ;CHECK THE INTERRUPT WITH A NOOP.  WE WILL SET UP THE
4549                                  ;INTERRUPT VECTOR, LOWER THE PSW TO ZERO AND ISSUE
4550                                  ;A NOOP.  THE INTERRUPT SERVICE ROUTINE WILL SET A
4551                                  ;FLAG UPON INTERRUPT AND RETURN IN LINE.  WE WAIT 200 MILLISECONDS
4552                                  ;LOOKING FOR THAT FLAG TO BE SET BEFORE CALLING IT
4553                                  ;AN ERROR.  IF THE INTERRUPT SENDS US TO ANOTHER
4554                                  ;VECTOR ADDRESS THEN THE ERROR HANDLER WILL REPORT
4555                                  ;"TRAP TO XXXX FROM YYYY" AND RETURN  TO DIAG SUP MONITOR.  IF THE
4556                                  ;INTERRUPT GOES TO ABOVE 1000 WHO KNOWS WHAT WILL HAPPEN.
4557  016500                          STARS
 (2)                                  ;*****************************************************************
4558
4559
4560  016500  005737  002252                  TST     T.CNTLR
4561  016504  001426                          BEQ     99$
4562
4563  016506  005037  002172                  CLR     INTFLG          ;CLEAR INTERRUPT OCCRUANCE FLAG
4564  016512                                  SETPRI  #PRI00          ;SET PSW TO 0
 (3)  016512  012700  000000                  MOV     #PRI00,R0
 (3)  016516  104041                          EMT     C$SPRI
4565  016520  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
4566  016524  000100                          NOOP0!INTEN             ;NOOP AND INTERRUPT ENABLE
4567  016526  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
4568  016532  005737  002172                  TST     INTFLG          ;DID INTERRUPT OCCUR
4569  016536  001004                          BNE     2$              ;IF SO BRANCH
4570  016540                                  ERRDF   22.,EM13,ERR0
 (3)  016540  104462                          TRAP    T$ERCODE
 (5)  016542  000026                          .WORD   22
 (5)  016544  004707                          .WORD   EM13
 (5)  016546  007556                          .WORD   ERR0
4571  016550  005037  002172          2$:     CLR     INTFLG
```

```
4572  016554                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  016554  104006                          EMT     C$CLP1
4573  016556  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
4574
4575
4576  016562                          99$:
4577  016562                          ENDTST                          ;****END OF TEST****
 (3)  016562                          L10045:
 (3)  016562  104001                          EMT     C$ETST
4578
4579
4580                                  .SBTTL  **TEST 24** - TEST PRIORITY BR LEVEL
4581
4582  016564                          BGNTST                          ;****START OF TEST****
4583
4584  016564                          STARS
 (2)                                  ;;*****************************************************************
4585                                  ;TEST THAT PRIORITY GIVEN IS ACTUAL PRIORITY OF CONTROLLER.  WE KNOW
4586                                  ;THE BOARD WILL INTERRUPT.  WE WILL START TRYING TO INTERRUPT AT 7
4587                                  ;AND WORK DOWN TIL IT DOES INTERRUPT.
4588  016564                          STARS
 (2)                                  ;;*****************************************************************
4589
4590  016564  005737  002252                  TST     T.CNTLR         ;RLV11??
4591  016570  001456                          BEQ     6$              ;YES, SKIP TEST
4592
4593  016572  012737  000340  002226          MOV     #340,BDDAT      ;SET UP INITIAL OF 7
4594  016600  013737  002132  002224          MOV     BPRIOR,GDDAT    ;GET GIVEN PRIORITY
4595
4596  016606                          BGNSEG                          ;****START OF SEGMENT****
 (3)  016606  104004                          EMT     C$BSEG
4597
4598  016610  005037  002172          5$:     CLR     INTFLG          ;CLEAR INTERRUPT OCCURRANCE
4599  016614                                  SETPRI  BDDAT           ;SET PRIORITY
 (3)  016614  013700  002226                  MOV     BDDAT,R0
 (3)  016620  104041                          EMT     C$SPRI
4600
4601  016622  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
4602  016626  000100                          NOOPO!INTEN
4603
4604  016630  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
4605  016634                          ESCAPE  TST                     ;IF /FL:LOE SET LOOP, ELSE EXIT TST
 (3)  016634  104010                          EMT     C$ESCAPE
 (3)  016636  000070                          .WORD   L10046-.
4606
4607  016640  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
4608  016644                          ESCAPE  TST                     ;IF /FL:LOE SET LOOP, ELSE EXIT TST
 (3)  016644  104010                          EMT     C$ESCAPE
 (3)  016646  000060                          .WORD   L10046-.
4609
4610  016650  023737  002226  002224          CMP     BDDAT,GDDAT     ;SHOULD IT INTERRUPT
4611  016656  002012                          BGE     1$              ;NO, BRANCH
4612
4613  016660  005737  002172                  TST     INTFLG          ;DID INTERRUPT OCCUR
4614  016664  001004                          BNE     2$              ;YES, OK
4615
```

```
                                3$:     ERRDF   204.,EM17,ERR7
 (3) 016666  104462                     TRAP    T$ERCODE
 (5) 016670  000314                     .WORD   204
 (5) 016672  005043                     .WORD   EM17
 (5) 016674  010044                     .WORD   ERR7
4617
4618 016676                     2$:     ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
 (3) 016676  104010                     EMT     C$ESCAPE
 (3) 016700  000014                     .WORD   10000$-.
4619 016702  000405                     BR      4$
4620 016704  005737  002172     1$:     TST     INTFLG          ;DID INTERRUPT OCCUR
4621 016710  001772                     BEQ     2$              ;NO, OK
4622 016712  000765                     BR      3$              ;YES, ERROR
4623
4624 016714                     ENDSEG                          ;****END OF SEGMENT****
 (3) 016714                     10000$:
 (3) 016714  104005                     EMT     C$ESEG
4625 016716  162737  000040  002226 4$:     SUB     #40,BDDAT       ;NEXT LEVEL
4626 016724  100331                     BPL     5$
4627
4628 016726                     6$:
4629 016726                     ENDTST                          ;****END OF TEST****
 (3) 016726                     L10046:
 (3) 016726  104001                     EMT     C$ETST
4630
4631                            .SBTTL  **TEST 25** - GET STATUS FUNCTION
4632
4633 016730                     BGNTST                          ;****START OF TEST****
4634
4635
4636 016730                     STARS
 (2)                            ;;*****************************************************************
4637                            ;TEST GET STATUS FUNCTION.  THE GET STATUS FUNCTION WILL
4638                            ;WORK IF DRIVE IS LOADED AND READY OR NOT.  THE RLDA
4639                            ;IS LOADED WITH THE GET STATUS AND MARKER BITS (BITS 1,0)
4640                            ;AND THE FUNCTION IS ISSUED.  WE WAIT 200 MILLISECONDS
4641                            ;FOR CONTROLLER READY.  VERIFY THAT NO ERRORS OCCUR.
4642 016730                     STARS
 (2)                            ;;*****************************************************************
4643
4644
4645 016730  012777  000013  163166         MOV     #GSBIT!MK!DRST,@RLDA    ;SET GET STATUS AND MARKER BI
4646 016736  004537  012552                 JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
4647 016742  000004                         GSTAT                   ;GET STATUS
4648 016744  004537  013362                 JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
4649 016750                     2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3) 016750  104006                     EMT     C$CLP1
4650
4651 016752  004537  012252                 JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
4652
4653 016756                     ENDTST                          ;****END OF TEST****
 (3) 016756                     L10047:
 (3) 016756  104001                     EMT     C$ETST
4654
4655
4656                            .SBTTL  **TEST 26** - GET STATUS FUNCTION INTERRUPT
```

```
OUTERR  MACY11 30.1046(  30-OCT-77  16:51  PAGE 85-17
DZRLAA.P11    05-OCT-77 10:41          **TEST 26** - GET STATUS FUNCTION INTERRUPT                    SEQ 0069

                                       BGNTST                          ;****START OF TEST****
                                       ;CHECK GET STATUS UNDER INTERRUPT
                                               CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE
                                               SETPRI  #PRI00          ;PSW TO LEVEL 0
                                               MOV     #PRI00,R0
                                               EMT     C$SPRI
                                               MOV     #GSBIT!MK,@RLDA ;SET UP DA
                                               JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                               GSTAT!INTEN             ;GET STATUS, INT ENABLE
                                               JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                               SETPRI  #PRI07
                                               MOV     #PRI07,R0
                                               EMT     C$SPRI
                                               TST     INTFLG          ;DID INTERRUPT OCCUR
                                               BNE     2$              ;YES-BRANCH
                                               ERRDF   28.,EM30,ERR0
                                               TRAP    T$ERCODE
                                               .WORD   28
                                               .WORD   EM30
                                               .WORD   ERR0
                                       2$:     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
```

```
CHERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 85-18
DERLAA.P11    05-OCT-77 10:41          **TEST 27** - GET STATUS FUNCTION GENERATES OPI W/O GS BIT                SEQ 0079

                                1$:     JSR     R5,CHERR
                                        CKLOOP
                                        EMT     C$CLP1
                                        BIT     #OPI.E.CS       ;IS OPI SET?
                                        BNE     2$              ;YES-BRANCH NO-CHECK TIMEOUT
                                        ERRDF   29.,EM33,ERR0
                                        TRAP    T$ERCODE
                                        .WORD   29
                                        .WORD   EM33
                                        .WORD   ERR0
                                2$:

                                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,ERR0
                                        TRAP    T$ERCODE
                                        .WORD   30
                                        .WORD   EM11
                                        .WORD   ERR0

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

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

```
                        JSR     R5,CHERR
                1$:     CKLOOP
                        EMT     C$CLP1
017236  104006          BIT     #OPI,E.CS       ;IS OPI SET?
017240  032737  002000  002152  BNE     3$              ;YES-BRANCH NO-CHECK TIMEOUT
017246  001004          ERRDF   31.,EM33,ERR0
017250                  TRAP    T$ERCODE
017250  104462          .WORD   31
017252  000037          .WORD   EM33
017254  005171          .WORD   ERR0
017256  007556
017260                  3$:


017260                  ENDTST                          ;****END OF TEST****
017260                  L10052:
017260  104001                  EMT     C$ETST

                        .SBTTL  **TEST 29** - READ HEADER FUNCTION

017262                  BGNTST                          ;****START OF TEST****
017262                  STARS
                        ;;*************************************************************
                        ;CHECK THAT READ HEADER WORKS, THAT WE CAN ISSUE
                        ;IT, GET READY BACK WITHOUT ANY ERRORS SETTING.
017262                  STARS
                        ;;*************************************************************


017262  004537  012552          JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
017266  000010                  RDHDR                   ;READ HEADER
017270  004537  013362          JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                                        ;READY
017274                  2$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
017274  104006                  EMT     C$CLP1

017276  004537  012252          JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS

017302                  ENDTST                          ;****END OF TEST****
017302                  L10053:
017302  104001                  EMT     C$ETST
```

4769

```
C.TERR  MACY11 30.1046(  30-OCT-77  16:51  PAGE 87
C2ERLAA.P11    05-OCT-77 10:41          **TEST 30** - READ HEADER FUNCTION INTERRUPT                SEQ 0073

                                        .SBTTL  **TEST 30** - READ HEADER FUNCTION INTERRUPT

017304                                  BGNTST                          ;****START OF TEST****

017304                                  STARS
                                        ;;*************************************************************
                                        ;CHECK THAT READ HEADER WILL GENERATE AN INTERRRUPT
                                        ;UPON COMPLETION WITHOUT ANY ERRORS SETTING
017304                                  STARS
                                        ;;*************************************************************

017304                                          SETPRI  #PRI00          ;PSW TO 0
017304  012700  000000                          MOV     #PRI00,R0
017310  104041                                  EMT     C$SPRI
017312  005037  002172                          CLR     INTFLG          ;CLEAR INTERRUPT OCCURANCE
017316  004537  012552                          JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
017322  000110                                  RDHDR!INTEN             ;READ HEADER, INTR. ENA
017324  004537  013362                          JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
017330                                          SETPRI  #PRI07
017330  012700  000340                          MOV     #PRI07,R0
017334  104041                                  EMT     C$SPRI
017336  005737  002172                          TST     INTFLG          ;INTERRUPT HAPPEN
017342  001004                                  BNE     2$              ;YES-CONTINUE
017344                                          ERRDF   35.,EM37,ERR0
017344  104462                                  TRAP    T$ERCODE
017346  000043                                  .WORD   35
017350  005313                                  .WORD   EM37
017352  007556                                  .WORD   ERR0
017354                                  2$:     CKLOOP                  ;CHECK IF  FL:LOE IS SET
017354  104006                                  EMT     C$CLP1
```

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

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

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

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

                                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.
                                STARS
                                ;;*************************************************************

                                                MOV     #100.,R1        ;SET UP TO DO 100 RD HDR'S
                                                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                                RDHDR                   ;READ HEADER
                                                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                99$:            ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
                                                EMT     C$ESCAPE
                                                .WORD   L10055-.

                                                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                                ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
                                                EMT     C$ESCAPE
                                                .WORD   L10055-.

                                                MOV     E.MP,GDDAT      ;READ FIRST HEADER (ASSUME GOOD)
                                                BIC     SECMSK,GDDAT    ;MASK AWAY SECTOR BITS
                                BGNSEG                                  ;****START OF SEGMENT****
                                                EMT     C$BSEG
                                2$:
                                                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                                RDHDR
                                                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                97$:            ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
                                                EMT     C$ESCAPE
                                                .WORD   10000$-.

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

                                                MOV     E.MP,BDDAT      ;READ HEADER
                                                BIC     SECMSK,BDDAT    ;MASK AWAY SECTOR BITS
                                                CMP     GDDAT,BDDAT     ;IS HEADER CORRECT
                                                BEQ     4$
```

```
                                        ERRDF   36.,EM41,ERR4
                                        TRAP    T$ERCODE
                                        .WORD   36
                                        .WORD   EM41
                                        .WORD   ERR4
                                                        ;CONSTANT CYL & HS
                                4$:     CKLOOP          ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1
                                        DEC     R1      ;PERFORM ALL READ HDR'S
                                        BNE     2$      ;IF NOT GO BACK AND DO ANOTHER
                                ENDSEG                  ;****END OF SEGMENT****
                                10000$:
                                        EMT     C$ESEG
                                ENDTST                  ;****END OF TEST****
                                L10055:
                                        EMT     C$ETST


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

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

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

                                        CLR     3$
                                        JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                        RDHDR                   ;READ HEADER
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
                                        EMT     C$ESCAPE
                                        .WORD   L10056-.

                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                        ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
                                        EMT     C$ESCAPE
                                        .WORD   L10056-.

                                        MOV     E.MP,2$         ;READ HEADER WORD
                                        JSR     R5,SIMBCC       ;GO CALCULATE CRC
                                        16.                     ;16 BITS
                                2$:     .WORD   0               ;HEADER GOES HERE
                                3$:     .WORD   0               ;START WITH 0 CRC
                                        MOV     CALBCC,5$
                                        MOV     E.MP1,4$        ;GET SECOND HALF
                                        JSR     R5,SIMBCC
                                        16.
                                4$:     .WORD   0
```

```
 4876  017626  000000                    5$:     .WORD   0
--- 4877  017630  013737  002206  002224          MOV     CALBCC,GDDAT    ;STORE CALCULATED CRC AS GOOD
 4878  017636  013737  002164  002226          MOV     E.MP2,BDDAT     ;THIRD READ OF DB GETS CRC
 4879  017644  023737  002224  002226          CMP     GDDAT,BDDAT     ;IS CRC CORRECT?
 4880  017652  001404                          BEQ     6$              ;IF SO CONTINUE
 4881
 4882  017654                                  ERRDF   37.,EM42,ERR4
  (3)  017654  104462                          TRAP    T$ERCODE
  (5)  017656  000045                          .WORD   37
  (5)  017660  005444                          .WORD   EM42
  (5)  017662  007722                          .WORD   ERR4
 4883  017664                            6$:
 4884
 4885  017664                            ENDTST                          ;****END OF TEST****
  (3)  017664                            L10056:
  (3)  017664  104001                          EMT     C$ETST
 4886
 4887
 4888                                    .SBTTL  **TEST 33** - CHECK CONSECUTIVE HEADERS
 4889
 4890  017666                            BGNTST                          ;****START OF TEST****
 4891
 4892
 4893  017666                            STARS
  (2)                                    ;;*****************************************************************
 4894                                    ;CHECK THAT THE HEADERS ARE CONSECUTIVE.  WE WILL DO
 4895                                    ;40 (FORTY) READ HEADERS AND STORE EACH.  AFTER WE HAVE
 4896                                    ;READ THE FORTIETH HEADER WE WILL VERIFY THAT
 4897                                    ;THEY CAME IN SEQUENTIAL, THAT 0 FOLLOWS 39,
 4898                                    ;THAT THERE WERE NO ERRORS.
 4899  017666                            STARS
  (2)                                    ;;*****************************************************************
 4900
 4901
 4902  017666  005037  002230                    CLR     FIRST           ;CLEAR FIRST READ DONE FLAG
 4903  017672  012703  002670                    MOV     #HDRBUF,R3      ;STORE HEADERS
 4904  017676  012701  000050                    MOV     #40.,R1         ;FOURTY HEADERS
 4905  017702  012737  000210  002140          MOV     #RDHDR!CRDY,B.CS
 4906  017710  053737  002136  002140          BIS     DRIVE,B.CS
 4907  017716  013777  002140  162174          MOV     B.CS,@RLCS
 4908  017724  042777  000200  162166  2$:     BIC     #200,@RLCS
 4909  017732  032777  000200  162160  1$:     BIT     #200,@RLCS      ;DONE?
 4910  017740  001774                          BEQ     1$
 4911  017742  017723  162152                  MOV     @RLCS,(R3)+
 4912  017746  017723  162154                  MOV     @RLMP,(R3)+
 4913  017752  017723  162150                  MOV     @RLMP,(R3)+
 4914  017756  017723  162144                  MOV     @RLMP,(R3)+
 4915  017762  005301                          DEC     R1              ;HAVE WE READ FOURTY HEADERS
 4916  017764  001357                          BNE     2$              ;GO BACK UNTIL FOURTY DONE
 4917  017766  012703  002670                  MOV     #HDRBUF,R3      ;GET LIST OF HEADERS
 4918  017772  012701  000050                  MOV     #40.,R1         ;CHECK FOURTY OF THEM
 4919  017776  011337  002152                  MOV     (R3),E.CS
 4920  020002  005737  002152                  TST     E.CS
 4921  020006  100016                          BPL     99$
 4922  020010  012737  003747  012534          MOV     #RHDMES,RESTMS
 4923  020016  005723                          TST     (R3)+
```

```
 4924  020020  012337  002160                 MOV     (R3)+,E.MP
 4925  020024  012337  002162                 MOV     (R3)+,E.MP1
 4926  020030  012337  002164                 MOV     (R3)+,E.MP2
 4927  020034  004537  012252                 JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
 4928  020040  000137  020202                 JMP     7$
 4929  020044  005723                 99$:    TST     (R3)+
 4930  020046  011337  002226                 MOV     (R3),BDDAT      ;GET HEADER
 4931  020052  005737  002230                 TST     FIRST           ;IS THIS FIRST READ?
 4932  020056  001007                         BNE     4$              ;NO, BRANCH
 4933  020060  012737  000001  002230         MOV     #1,FIRST        ;SET FIRST READ DONE FLAG
 4934  020066  013737  002226  002224  3$:    MOV     BDDAT,GDDAT     ;SET UP NEXT READ EXPECTED
 4935  020074  000435                         BR      6$              ;GO SEE IF TEST IS DONE
 4936  020076  005237  002224          4$:    INC     GDDAT           ;INCREMENT EXP'D HEADER
 4937  020102  023737  002226  002224         CMP     BDDAT,GDDAT     ;IS NEW HEADER SEQUENTIAL?
 4938  020110  001766                         BEQ     3$              ;YES THEN BRANCH
 4939  020112  033737  002176  002226         BIT     SECMSK,BDDAT    ;IS NEW HEADER ZERO?
 4940  020120  001015                         BNE     5$              ;NO, THEN ERROR GO REPORT IT
 4941  020122  013737  002224  002210         MOV     GDDAT,TEMP2     ;YES, CHECK IF LAST HEADER WAS
 4942  020130  043737  002232  002210         BIC     CYLMSK,TEMP2    ;MAX ADDRESS, IF SO BRANCH
 4943  020136  023737  002234  002210         CMP     MXSEC1,TEMP2    ;STORE NEW DATA AS OLD
 4944  020144  001750                         BEQ     3$              ;AND PERFORM NEW RD HDR
 4945  020146  043737  002176  002224         BIC     SECMSK,GDDAT    ;EXPECTING ZERO SECTOR
 4946
 4947  020154                         5$:
 4948
 4949  020154  005037  002230                 CLR     FIRST           ;ERROR WILL MAKE US MISS
 4950                                                                 ;NEXT SECTOR SEQUENTIALLY
 4951                                                                 ;START OVER; CLEAR FIRST FLAG
 4952  020160                                 ERRDF   38.,EM43,ERR2
   (3) 020160  104462                         TRAP    T$ERCODE
   (5) 020162  000046                         .WORD   38
   (5) 020164  005502                         .WORD   EM43
   (5) 020166  007606                         .WORD   ERR2
 4953  020170                         6$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
   (3) 020170  104006                         EMT     C$CLP1
 4954
 4955  020172  062703  000006                 ADD     #6,R3
 4956  020176  005301                         DEC     R1              ;HAVE WE DONE THIS ENOUGH
 4957  020200  001321                         BNE     99$             ;NO, GO BACK DO IT AGAIN
 4958  020202                         7$:
 4959  020202                         ENDTST                          ;****END OF TEST****
   (3) 020202                         L10057:
   (3) 020202  104001                         EMT     C$ETST
 4960
 4961
 4962                                 .SBTTL  **TEST 34** - SEEK FUNCTION
 4963
 4964  020204                         BGNTST                          ;****START OF TEST****
 4965  020204                         STARS
   (2)                                ;;*****************************************************************
 4966                                 ;CHECK THE SEEK FUNCTION RETURNS CONTROLLER READY
 4967                                 ;WITH NO ERRORS.  WE ISSUE A ONE TRACK IN WORD SEEK.
 4968                                 ;WE DO NOT CHECK THE RESULT FOR POSITION
 4969  020204                         STARS
   (2)                                ;;*****************************************************************
 4970
```

```
                                                MOV     #BIT7!MK!SIGN,@RLDA  ;SET UP DA-DIFF=1,MARKER,TOWARDS
                                                JSR     R5,LDFUNC            ;ISSUE FUNCTION OF FOLLOWING WORD
                                                SEEK                         ;SEEK
                                                JSR     R5,WTCRDY            ;WAIT FOR CONTROLLER READY HIGH
                                                WAITMS  #2.
                                                MOV     #2.,R0
                                                EMT     C$WTM
                                        2$:     CKLOOP                       ;CHECK IF /FL:LOE IS SET
                                                EMT     C$CLP1

                                                JSR     R5,CHERR             ;CHECK CONTROLLER FOR ERRORS

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


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

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


                                        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
                                        STARS
                                        ;;*****************************************************************


                                                MOV     #BIT7!MK,@RLDA  ;SET DA, MARKER, DIFF=1.
                                                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

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

                                                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS

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

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

020306                          BGNTST                          ;****START OF TEST****


020306                          STARS
                                ;;*****************************************************************
                                ;CHECK THAT CONTROLLER READY RESETTING WHEN THE SEEK IS
                                ;INITIATED CAUSES AN INTERRUPT BUT DRIVE READY WILL
                                ;NOT.  WE ALSO MONITOR FOR ANY ERROR BITS SETTING.
020306                          STARS
                                ;;*****************************************************************


020306  005037  002172                  CLR     INTFLG
020312                          SETPRI  #PRI00                  ;SET PSW TO 0
020312  012700  000000                  MOV     #PRI00,R0
020316  104041                  EMT     C$SPRI
020320  012777  000205  161576          MOV     #BIT7!MK!SIGN,@RLDA     ;SET UP RLDA
020326  004537  012552                  JSR     R5,LDFUNC               ;ISSUE FUNCTION OF FOLLOWING WORD
020332  000106                  SEEK!INTEN                              ;SEEK AND INTR. ENA.
020334  004537  013362                  JSR     R5,WTCRDY               ;WAIT FOR CONTROLLER READY HIGH
020340  000240                  NOP                                     ;
020342  005737  002172          1$:     TST     INTFLG                  ;DID INTERRUPT OCCUR
020346  001004                  BNE     2$                              ;YES, GO CHECK DRDY
020350                          ERRDF   40.,EM47,ERR0
020350  104462                  TRAP    T$ERCODE
020352  000050                  .WORD   40
020354  005722                  .WORD   EM47
020356  007556                  .WORD   ERR0
020360                          2$:     CKLOOP                          ;CHECK IF /FL:LOE IS SET
020360  104006                  EMT     C$CLP1


020362  004537  012252                  JSR     R5,CHERR                ;CHECK CONTROLLER FOR ERRORS
020366                          CKLOOP                                  ;CHECK IF /FL:LOE IS SET
020366  104006                  EMT     C$CLP1

020370  005037  002172                  CLR     INTFLG                  ;CLEAR INTERRUPT OCCURANCE


020374  004537  013316                  JSR     R5,WTDRDY               ;WAIT FOR DRIVE READY
020400                          5$:     CKLOOP                          ;CHECK IF  FL:LOE IS SET
020400  104006                  EMT     C$CLP1

020402                          SETPRI  #PRI07
020402  012700  000340                  MOV     #PRI07,R0
020406  104041                  EMT     C$SPRI
020410  005737  002172                  TST     INTFLG                  ;DID DRIVE READY CAUSE INTERRUPT
020414  001404                  BEQ     6$                              ;NO, CONTINUE

020416                          ERRDF   42.,EM52,ERR0
020416  104462                  TRAP    T$ERCODE
020420  000052                  .WORD   42
```

```
                (5)  020422  005753                        .WORD   EM52
                (5)  020424  007556                        .WORD   ERR0
5056            020426                          6$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                (3)  020426  104006                        EMT     C$CLP1
5057
5058            020430                          ENDTST                          ;****END OF TEST****
                (3)  020430                     L10062:
                (3)  020430  104001                        EMT     C$ETST
5059
5060
5061                                            .SBTTL  **TEST 37** - TEST DIFFERENCE WORD TRANSMISSION
5062
5063            020432                          BGNTST                          ;****START OF TEST****
5064
5065
5066
5067
5068            020432                          STARS
                (2)                             ;;*****************************************************************
5069                                            ;VERIFY THAT THE DIFFERENCE WORD LOADS AND IS
5070                                            ;TRANSMITTED CORRECTLY.  WE WILL ISSUE SEEKS WITH THE
5071                                            ;DIFFERENCE WORD CONTAINING ALL OF THE BIT PATTERNS FLOATING 1,
5072                                            ;GROWING 1, GROWING 0 AND SHITING 0.  THE SEEK WILL
5073                                            ;START FROM TRACK 0 EACH TIME AND WILL RETURN THERE
5074                                            ;EACH, THUS BOTH DIRECTIONS FOR PATTERNS WILL BE CHECKED.
5075                                            ;READ HEADERS ARE USED TO VERIFY THE SEEK CORRECTNESS.
5076                                            ;ERRORS ARE MONITORED AND REPORTED.
5077            020432                          STARS
                (2)                             ;;*****************************************************************
5078
5079
5080            020432  012703  002466                    MOV     #SKLST,R3       ;GET LIST OF DIFFERENCE WORDS
5081            020436                          BGNSEG                          ;****START OF SEGMENT****
                (3)  020436  104004                        EMT     C$BSEG
5082            020440                          1$:
5083            020440  004537  012552                    JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5084            020444  000010                            RDHDR                   ;READ HEADER
5085            020446  004537  013362                    JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5086            020452                          98$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
                (3)  020452  104006                        EMT     C$CLP1
5087
5088            020454  004537  012252                    JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5089            020460                                    CKLOOP                  ;CHECK IF /FL:LOE IS SET
                (3)  020460  104006                        EMT     C$CLP1
5090
5091            020462  013737  002160  002226            MOV     E.MP,BDDAT      ;READ HEADER
5092            020470  043737  002176  002226            BIC     SECMSK,BDDAT    ;CLEAR OUT SECTOR
5093            020476  001462                            BEQ     99$             ;IF ON TRACK ZERO, H.S. ZERO. OK
5094
5095                                            ;NOT ON TRACK ZERO CALCULATE DIFFERENCE WORD AND PUT IT BACK
5096                                            ;ON ZERO.
5097
5098            020500  042737  000100  002226            BIC     #RHHS,BDDAT     ;CLEAR OUT HEAD SELECT
5099            020506  013777  002226  161410            MOV     BDDAT,@RLDA     ;PUT CYLINDER AS DIFFERENCE WORD
5100            020514  052777  000001  161402            BIS     #MK,@RLDA       ;SET MARKER BIT
5101            020522  004537  012552                    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

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

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

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

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

                                                CLR     GDDAT           ;CLEAR EXPECTED
                                                MOV     BDDAT,DWORD     ;SAVE DIFFERENCE WORD
                                                MOV     E.MP,BDDAT      ;READ HEADER
                                                BIC     SECMSK,BDDAT    ;MASK OUT SECTOR BITS
                                                BEQ     5$              ;BRANCH IF ON ZERO TRACK

                                                ERRDF   43.,EM54,ERR3
                                                TRAP    T$ERCODE
                                                .WORD   43
                                                .WORD   EM54
                                                .WORD   ERR3
                                        5$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                                EMT     C$CLP1

                                        99$:    MOV     (R3),@RLDA      ;GET DIFFERENCE WORD
                                                BIS     #SIGN!MK,@RLDA  ;SET SIGN (TOWARDS SPINDLE) AND MARKER
                                                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                                SEEK                    ;SEEK
                                                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                                EMT     C$CLP1

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

                                                JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
                                        87$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                                EMT     C$CLP1
```

```
                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                        CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1
                                        JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
                                        RDHDR                   ;READ HEADER
                                        JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
                                        CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1
                                        JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
                                        ESCAPE  SEG             ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
                                        EMT     C$ESCAPE
                                        .WORD   10000$-.
                                        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
                                        TRAP    T$ERCODE
                                        .WORD   44
                                        .WORD   EM54
                                        .WORD   ERR3
                                9$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
                                        EMT     C$CLP1
                                        TST     (R3)+           ;BUMP PATTERN
                                        CMP     R3,#SKEND       ;DID WE DO ALL PATTERNS?
                                        BEQ     10$             ;YES, GO TO NEXT TEST
                                        JMP     1$              ;NO, GO BACK WITH NEXT PATTERN

                                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
```

```
OUTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 88-9
DZRLAA.P11    05-OCT-77 10:41          **TEST 38** - VERIFY HEAD SELECT 0 VIA RD HDR                    SEQ 0083

                                       ;SEEK TO HEAD SELECT 0 AND VERIFY WITH READ HEADER.
                                       STARS
                                       ;;*****************************************************************

                                       ;SET MARKER IN RLDA
                                       ;SET EXPECTED
                                       ;LOAD HS=0 INTO RLDA

                                       ;ISSUE FUNCTION OF FOLLOWING WORD
                                       ;SEEK
                                       ;WAIT FOR CONTROLLER READY HIGH
                                       ;CHECK IF /FL:LOE IS SET

                                       ;CHECK CONTROLLER FOR ERRORS
                                       ;CHECK IF /FL:LOE IS SET

                                       ;WAIT FOR DRIVE READY
                                       ;CHECK IF /FL:LOE IS SET

                                       ;CHECK CONTROLLER FOR ERRORS
                                       ;CHECK IF /FL:LOE IS SET

                                       ;ISSUE FUNCTION OF FOLLOWING WORD
                                       ;READ HEADER
                                       ;WAIT FOR CONTROLLER READY HIGH
                                       ;CHECK IF /FL:LOE IS SET

                                       ;CHECK CONTROLLER FOR ERRORS
                                       ;IF /FL:LOE SET LOOP, ELSE EXIT TST

                                       ;READ HEADER FOR HEAD SELECT
                                       ;MASK ONLY HEAD SELECT
                                       ;COMPARE HEAD SELECTS
                                       ;IF EQUAL CONTINUE

                                       ;****END OF TEST****

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

```
5228  021160                          BGNTST                          ;****START OF TEST****
5229
5230
5231  021160                          STARS
 (2)                                  ;;*****************************************************************
5232                                  ;CHECK THAT WE CAN SELECT HEAD SELECT ONE.  ISSUE
5233                                  ;SEEK TO HEAD SELECT 1 AND VERIFY WITH READ HEADER.
5234  021160                          STARS
 (2)                                  ;;*****************************************************************
5235
5236
5237  021160  012777  000001  160736  99$:    MOV     #MK,@RLDA       ;SET MARKER IN RLDA
5238  021166  052777  000020  160730          BIS     #DAHS,@RLDA     ;LOAD HS=1 INTO RLDA
5239  021174  004537  012552          2$:     JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5240  021200  000006                          SEEK                    ;SEEK
5241  021202  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5242  021206                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  021206  104006                          EMT     C$CLP1
5243
5244  021210  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5245  021214                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  021214  104006                          EMT     C$CLP1
5246
5247  021216  004537  013316                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE CLEAR
5248  021222                          89$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  021222  104006                          EMT     C$CLP1
5249
5250  021224  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5251  021230                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  021230  104006                          EMT     C$CLP1
5252
5253  021232  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5254  021236  000010                          RDHDR                   ;READ HEADER
5255  021240  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5256  021244                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  021244  104006                          EMT     C$CLP1
5257
5258  021246  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5259  021252                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
 (3)  021252  104010                          EMT     C$ESCAPE
 (3)  021254  000044                          .WORD   L10065-.
5260
5261  021256  013737  002160  002226          MOV     E.MP,BDDAT      ;READ HEADER
5262  021264  042737  177677  002226          BIC     #177677,BDDAT   ;MASK FOR H.S.
5263  021272  012737  000100  002224          MOV     #RHHS,GDDAT     ;SET EXPECTED
5264  021300  023737  002224  002226          CMP     GDDAT,BDDAT     ;CORRECT HEAD
5265  021306  001404                          BEQ     5$              ;YES, CONTINUE
5266
5267  021310                                  ERRDF   46.,EM55,ERR4
 (3)  021310  104462                          TRAP    T$ERCODE
 (5)  021312  000056                          .WORD   46
 (5)  021314  006062                          .WORD   EM55
 (5)  021316  007722                          .WORD   ERR4
5268  021320                          5$:
5269
5270  021320                          ENDTST                          ;****END OF TEST****
```

```
OUTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 88-11
DZRLAA.P11    05-OCT-77 10:41        **TEST 39** - VERIFY HEAD SELECT 1 VIA RD HDR                                SEQ 0085

 (3)  021320                         L10065:
 (3)  021320  104001                         EMT     C$ETST
5277
5278
5279
5280                                 .SBTTL  **TEST 40** - VERIFY HEAD SELECT 0 VIA GET STATUS
5281  021322                         BGNTST                              ;****START OF TEST****
5282
5283  021322                         STARS
 (2)                                 :;*****************************************************************
5284                                 :CHECK THAT WE CAN READ BACK HEAD SELECT 0 WITH
5285                                 :A GET STATUS FUNCTION.  SELECT H.S. 0 WITH A SEEK
5286                                 :VERIFY WITH GET STATUS
5287  021322                         STARS
 (2)                                 :;*****************************************************************
5288
5289  021322  012777  000001  160574         MOV     #MK,@RLDA           ;SET MARKER IN RLDA
5290                                                                      ;LOAD HS=0 INTO RLDA
5291  021330  005037  002224         2$:     CLR     GDDAT               ;SET UP EXP'D
5292  021334  004537  012552         3$:     JSR     R5,LDFUNC           ;ISSUE FUNCTION OF FOLLOWING WORD
5293  021340  000006                         SEEK                        ;SEEK
5294  021342  004537  013362                 JSR     R5,WTCRDY           ;WAIT FOR CONTROLLER READY HIGH
5295  021346                                 CKLOOP                      ;CHECK IF /FL:LOE IS SET
 (3)  021346  104006                         EMT     C$CLP1
5296
5297  021350  004537  012252                 JSR     R5,CHERR            ;CHECK CONTROLLER FOR ERRORS
5298  021354                                 CKLOOP                      ;CHECK IF /FL:LOE IS SET
 (3)  021354  104006                         EMT     C$CLP1
5299
5300  021356  004537  013316                 JSR     R5,WTDRDY           ;WAIT FOR DRIVE READY
5301  021362                                 CKLOOP                      ;CHECK IF /FL:LOE IS SET
 (3)  021362  104006                         EMT     C$CLP1
5302
5303  021364  004537  012252                 JSR     R5,CHERR            ;CHECK CONTROLLER FOR ERRORS
5304  021370                                 CKLOOP                      ;CHECK IF /FL:LOE IS SET
 (3)  021370  104006                         EMT     C$CLP1
5305
5306  021372  012777  000003  160524         MOV     #GSBIT!MK,@RLDA     ;SET UP FOR GET STATUS IN DA
5307  021400  004537  012552                 JSR     R5,LDFUNC           ;ISSUE FUNCTION OF FOLLOWING WORD
5308  021404  000004                         GSTAT                       ;GET STATUS
5309  021406  004537  013362                 JSR     R5,WTCRDY           ;WAIT FOR CONTROLLER READY HIGH
5310  021412                                 CKLOOP                      ;CHECK IF /FL:LOE IS SET
 (3)  021412  104006                         EMT     C$CLP1
5311
5312  021414  004537  012252                 JSR     R5,CHERR            ;CHECK CONTROLLER FOR ERRORS
5313  021420                                 ESCAPE  TST                 ;IF /FL:LOE SET LOOP, ELSE EXIT TST
 (3)  021420  104010                         EMT     C$ESCAPE
 (3)  021422  000036                         .WORD   L10066-.
5314
5315  021424  013737  002160  002226         MOV     E.MP,BDDAT          ;READ STATUS FOR HEAD SELECT BIT
5316  021432  042737  177677  002226         BIC     #177677,BDDAT       ;LEAVE ONLY H.S. BIT
5317  021440  023737  002224  002226         CMP     GDDAT,BDDAT         ;IS HEAD SELECT CORRECT?
5318  021446  001404                         BEQ     6$                  ;YES, CONTINUE
5319
 (4)  021450                         ERRDF   47.,EMS6,ERR4
 (3)  021450  104462                TRAP    T$ERCODE
```

```
                021452  000057                          .WORD   47
                021454  006115                          .WORD   EMS6
                021456  007722                          .WORD   ERR4
 5315           021460                          6$:
 5316
 5317           021460                          ENDTST                          ;****END OF TEST****
                021460                          L10066:
                021460  104001                          EMT     C$ETST
 5318
 5319
 5320                                           .SBTTL  **TEST 41** - VERIFY HEAD SELECT 1 VIA GET STATUS
 5321
 5322           021462                          BGNTST                          ;****START OF TEST****
 5323
 5324           021462                          STARS
                                                ;;*****************************************************************
 5325                                           ;CHECK THAT WE CAN READ BACK HEAD SELECT 1 WITH A GET
 5326                                           ;STATUS FUNCTION.  SELECT H.S. 1 WITH A SEEK AND VERIFIY WITH
 5327                                           ;GET STATUS
 5328           021462                          STARS
                                                ;;*****************************************************************
 5329
 5330
 5331   021462  012777  000001  160434          MOV     #MK,@RLDA       ;SET MARKER IN RLDA
 5332   021470  052777  000020  160426          BIS     #DAHS,@RLDA     ;LOAD HS=1 INTO RLDA
 5333   021476  012737  000100  002224  2$:     MOV     #STHS,GDDAT     ;SET UP EXP'D
 5334   021504  004537  012552          3$:     JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
 5335   021510  000006                          SEEK                    ;SEEK
 5336   021512  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
 5337   021516                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
                021516  104006                  EMT     C$CLP1
 5338
 5339   021520  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
 5340   021524                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
                021524  104006                  EMT     C$CLP1
 5341
 5342   021526  004537  013316                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
 5343   021532                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
                021532  104006                  EMT     C$CLP1
 5344
 5345   021534  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
 5346   021540                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
                021540  104006                  EMT     C$CLP1
 5347
 5348   021542  012777  000003  160354          MOV     #GSBIT!MK,@RLDA ;SET UP FOR GET STATUS IN DA
 5349   021550  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
 5350   021554  000004                          GSTAT                   ;GET STATUS
 5351   021556  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
 5352   021562                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
                021562  104010                  EMT     C$ESCAPE
                021564  000046                  .WORD   L10067-.
 5353
 5354   021566  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
 5355   021572                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
                021572  104010                  EMT     C$ESCAPE
                021574  000036                  .WORD   L10067-.
```

```
5356
5357  021576  013737  002160  002226          MOV     E.MP,BDDAT      ;READ STATUS FOR HEAD SELECT BIT
5358  021604  042737  177677  002226          BIC     #177677,BDDAT   ;LEAVE ONLY H.S. BIT
5359  021612  023737  002224  002226          CMP     GDDAT,BDDAT     ;IS HEAD SELECT CORRECT?
5360  021620  001404                          BEQ     6$              ;YES, CONTINUE
5361
5362  021622                                  ERRDF   48.,EM56,ERR4
 (3)  021622  104462                          TRAP    T$ERCODE
 (5)  021624  000060                          .WORD   48
 (5)  021626  006115                          .WORD   EM56
 (5)  021630  007722                          .WORD   ERR4
5363  021632                          6$:
5364
5365  021632                          ENDTST                          ;****END OF TEST****
 (3)  021632                          L10067:
 (3)  021632  104001                          EMT     C$ETST
5366
5367
5368                                  .SBTTL  **TEST 42** - TEST TIME AT WHICH DIF WD GETS TRANSMITTED
5369
5370  021634                          BGNTST                          ;****START OF TEST****
5371
5372
5373  021634                          STARS
 (2)                                  ;;*****************************************************************
5374                                  ;VERIFY THAT THE DIFFERENCE WORD ON A SEEK IS
5375                                  ;TRANSMITTED PRIOR TO CONTROLLER READY SETTING.  THIS
5376                                  ;IS DONE BY SETTING A KNOWN DIFFERENCE WORD IN
5377                                  ;THE RLDA ISSUING A A SEEK, WAITING FOR CONTROLLER READY
5378                                  ;(BUT NOT DRIVE READY), WRITING A DIFFERENT RLDA AND WAITING
5379                                  ;FOR DRIVE READY.  THE RESULTANT POSITION SHOULD BE THAT
5380                                  ;OF THE FIRST RLDA ONLY.
5381  021634                          STARS
 (2)                                  ;;*****************************************************************
5382
5383
5384  021634  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5385  021640  000010                          RDHDR                   ;READ HEADER
5386  021642  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5387  021646                          99$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  021646  104006                          EMT     C$CLP1
5388
5389  021650  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5390  021654                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  021654  104006                          EMT     C$CLP1
5391
5392  021656  013737  002160  002224          MOV     E.MP,GDDAT      ;READ HEADER
5393  021664  043737  002176  002224          BIC     SECMSK,GDDAT    ;CLEAR SECTOR BITS
5394  021672  012777  000001  160224          MOV     #MK,@RLDA       ;SET MARKER IN RLDA
5395  021700  032737  000100  002224          BIT     #RHHS,GDDAT     ;TEST H.S.
5396  021706  001403                          BEQ     2$              ;IF ZERO, CONTINUE
5397  021710  052777  000020  160206          BIS     #DAHS,@RLDA     ;ONE, SET SO WE WILL REMAIN THERE
5398  021716  013737  002224  002216  2$:     MOV     GDDAT,TMP0      ;STORE HEADER
5399  021724  042737  000100  002216          BIC     #RHHS,TMP0      ;CLEAR H.S. FROM STORED WORD
5400  021732  023737  002216  002544          CMP     TMP0,HALMAX     ;WHERE ARE WE?
5401  021740  101007                          BHI     3$              ;BRANCH IF ON INNER HALF
```

```
 5402  021742  052777  000004  160154          BIS     #SIGN,@RLDA     ;ON OUTER HALF, SET SEEK TO GO IN
 5403  021750  063737  002542  002224          ADD     QUAMAX,GDDAT    ;SET UP EXPECTED HEADER
 5404  021756  000403                          BR      4$              ;CONTINUE
 5405  021760  163737  002542  002224  3$:     SUB     QUAMAX,GDDAT    ;SET UP EXPECTED HEADER
 5406  021766  053777  002542  160130  4$:     BIS     QUAMAX,@RLDA    ;SET DIFFERENCE WORD IN RLDA
 5407  021774  012737  000001  002220          MOV     #MK,TMP1        ;SET UP ANOTHER "RLDA" FOR LOADING
 5408  022002  032777  000020  160114          BIT     #DAHS,@RLDA     ;AFTER SEEK, TO CHANGE ONLY
 5409  022010  001003                          BNE     5$              ;HEAD
 5410  022012  052737  000020  002220          BIS     #DAHS,TMP1
 5411  022020  004537  012552          5$:     JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
 5412  022024  000006                          SEEK                    ;SEEK
 5413  022026  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
 5414  022032                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
   (3) 022032  104006                          EMT     C$CLP1
 5415
 5416
 5417  022034  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
 5418  022040                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
   (3) 022040  104006                          EMT     C$CLP1
 5419
 5420  022042  013777  002220  160054          MOV     TMP1,@RLDA      ;SEND IN NEW DIFFERENCE WORD
 5421  022050  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
 5422  022054                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
   (3) 022054  104006                          EMT     C$CLP1
 5423
 5424  022056  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
 5425  022062                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
   (3) 022062  104006                          EMT     C$CLP1
 5426
 5427  022064  004537  013316                  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
 5428  022070                          8$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
   (3) 022070  104006                          EMT     C$CLP1
 5429
 5430
 5431  022072  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
 5432  022076                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
   (3) 022076  104006                          EMT     C$CLP1
 5433
 5434  022100  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
 5435  022104  000010                          RDHDR                   ;READ HEADER
 5436  022106  004537  013362                  JSR     R5;WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
 5437  022112                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
   (3) 022112  104006                          EMT     C$CLP1
 5438
 5439  022114  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
 5440  022120                                  ESCAPE  TST             ;IF /FL:LOE SET LOOP, ELSE EXIT TST
   (3) 022120  104010                          EMT     C$ESCAPE
   (3) 022122  000036                          .WORD   L10070-.
 5441
 5442  022124  013737  002160  002226          MOV     E.MP,BDDAT      ;READ HEADER
 5443  022132  043737  002176  002226          BIC     SECMSK,BDDAT    ;CLEAR SECTOR ADDRESS
 5444  022140  023737  002224  002226          CMP     GDDAT,BDDAT     ;IS HEADER CORRECT?
 5445  022146  001404                          BEQ     10$             ;IF SO BRANCH
 5446
 5447  022150                                  ERRDF   50.,EM57,ERR4
   (3) 022150  104462                          TRAP    T$ERCODE
```

```
OLTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 88-15
DZRLAA.P11    05-OCT-77 10:41          **TEST 42** - TEST TIME AT WHICH DIF WD GETS TRANSMITTED                SEQ 0089

 (5)  022152  000062                          .WORD   50
 (5)  022154  006154                          .WORD   EMS7
 (5)  022156  007722                          .WORD   ERR4
5448  022160                          10$:
5449
5450  022160                          ENDTST                          ;****END OF TEST****
 (3)  022160                          L10070:
 (3)  022160  104001                          EMT     C$ETST
5451
5452
5453                                  .SBTTL  **TEST 43** - EXTENSIVE CHECK OF HEADER CRC
5454
5455  022162                          BGNTST                          ;****START OF TEST****
5456  022162                          STARS
 (2)                                  ;;**************************************************************
5457                                  ;MORE EXTENSIVE CHECK OF HEADER CRC.  WE WILL SEEK
5458                                  ;AND READ HEADERS VERIFYING HDR CRC ACROSS THE
5459                                  ;PLATTER USING THE GROWING 0, GROWING 1, SHIFTING 0 AND
5460                                  ;GROWING 0 PATTERNS FOR TRACK ADDRESSES.
5461  022162                          STARS
 (2)                                  ;;**************************************************************
5462
5463
5464  022162  012703  002466                  MOV     #SKLST,R3       ;GET LIST OF DIFFERENCE WORDS
5465  022166                          BGNSEG                          ;****START OF SEGMENT****
 (3)  022166  104004                          EMT     C$BSEG
5466  022170                          1$:
5467  022170  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5468  022174  000010                          RDHDR                   ;READ HEADER
5469  022176  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5470  022202                          98$:    CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022202  104006                          EMT     C$CLP1
5471
5472  022204  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5473  022210                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022210  104006                          EMT     C$CLP1
5474
5475  022212  013737  002160  002226          MOV     E.MP,BDDAT      ;READ HEADER
5476  022220  043737  002176  002226          BIC     SECMSK,BDDAT    ;CLEAR OUT SECTOR
5477  022226  001461                          BEQ     5$              ;IF ON TRACK ZERO, H.S. ZERO, Ok
5478
5479                                  ;NOT ON TRACK ZERO CALCULATE DIFFERENCE WORD AND PUT IT BACK
5480                                  ;ON ZERO.
5481
5482  022230  042737  000100  002226          BIC     #RHHS,BDDAT     ;CLEAR OUT HEAD SELECT
5483  022236  013777  002226  157660          MOV     BDDAT,@RLDA     ;PUT CYLINDER AS DIFFERENCE WORD
5484  022244  052777  000001  157652          BIS     #MK,@RLDA       ;SET MARKER BIT
5485  022252  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5486  022256  000006                          SEEK                    ;SEEK
5487  022260  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5488  022264                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022264  104006                          EMT     C$CLP1
5489
5490  022266  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5491  022272                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022272  104006                          EMT     C$CLP1
```

```
5492
5493  022274  004537  013316          89$:  JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
5494  022300                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022300  104006                        EMT     C$CLP1
5495
5496  022302  004537  012252                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5497  022306                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022306  104006                        EMT     C$CLP1
5498
5499
5500  022310  004537  012552                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5501  022314  000010                        RDHDR                   ;READ HEADER
5502  022316  004537  013362                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5503  022322                          96$:  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022322  104006                        EMT     C$CLP1
5504
5505  022324  004537  012252                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5506  022330                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022330  104006                        EMT     C$CLP1
5507
5508  022332  005037  002224                CLR     GDDAT           ;CLEAR EXPECTED
5509  022336  013737  002226  002240        MOV     BDDAT,DWORD     ;SAVE DIFFERENCE WORD
5510  022344  013737  002160  002226        MOV     E.MP,BDDAT      ;READ HEADER
5511  022352  043737  002176  002226        BIC     SECMSK,BDDAT    ;MASK OUT SECTOR BITS
5512  022360  001404                        BEQ     5$              ;BRANCH IF ON ZERO TRACK
5513
5514  022362                                ERRDF   51.,EM54,ERR3
 (3)  022362  104462                        TRAP    T$ERCODE
 (5)  022364  000063                        .WORD   51
 (5)  022366  006023                        .WORD   EM54
 (5)  022370  007650                        .WORD   ERR3
5515  022372                           5$:  CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022372  104006                        EMT     C$CLP1
5516
5517  022374  011377  157524                MOV     (R3),@RLDA      ;GET DIFFERENCE WORD
5518  022400  052777  000005  157516        BIS     #SIGN!MK,@RLDA  ;SET SIGN (TOWARDS SPINDLE) AND MARKER
5519  022406  004537  012552                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5520  022412  000006                        SEEK                    ;SEEK
5521  022414  004537  013362                JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5522  022420                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022420  104006                        EMT     C$CLP1
5523
5524  022422  004537  012252                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5525  022426                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022426  104006                        EMT     C$CLP1
5526
5527  022430  004537  013316                JSR     R5,WTDRDY       ;WAIT FOR DRIVE READY
5528  022434                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022434  104006                        EMT     C$CLP1
5529
5530
5531  022436  004537  012252                JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5532  022442                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
 (3)  022442  104006                        EMT     C$CLP1
5533
5534  022444  004537  012552                JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
```

```
 5535  022450  000010                          RDHDR                           ;READ HEADER
 5536  022452  004537  013362                  JSR     R5,WTCRDY               ;WAIT FOR CONTROLLER READY HIGH
 5537  022456                                  CKLOOP                          ;CHECK IF /FL:LOE IS SET
   (3) 022456  104006                          EMT     C$CLP1
 5538
 5539
 5540  022460  004537  012252                  JSR     R5,CHERR                ;CHECK CONTROLLER FOR ERRORS
 5541  022464                                  CKLOOP                          ;CHECK IF /FL:LOE IS SET
   (3) 022464  104006                          EMT     C$CLP1
 5542
 5543  022466  011337  002224          8$:     MOV     (R3),GDDAT              ;GET EXPECTED CYLINDER
 5544  022472  011337  002240                  MOV     (R3),DWORD              ;SET UP DIFFERENCE FOR SEEK
 5545  022476  013737  002160  002226          MOV     E.MP,BDDAT              ;READ HEADER FROM RLMP
 5546  022504  043737  002176  002226          BIC     SECMSK,BDDAT            ;CLEAR OUT SECTOR BITS
 5547  022512  023737  002224  002226          CMP     GDDAT,BDDAT             ;DID SEEK GO TO THE RIGHT
 5548  022520  001404                          BEQ     9$                      ;TRACK, IF SO, GO GET NEXT
 5549
 5550  022522                                  ERRDF   52.,EM54,ERR3
   (3) 022522  104462                          TRAP    T$ERCODE
   (5) 022524  000064                          .WORD   52
   (5) 022526  006023                          .WORD   EM54
   (5) 022530  007650                          .WORD   ERR3
 5551  022532                          9$:     CKLOOP                          ;CHECK IF /FL:LOE IS SET
   (3) 022532  104006                          EMT     C$CLP1
 5552
 5553  022534  013737  002160  022550          MOV     E.MP,10$                ;GET HEADER WORD
 5554  022542  004537  013130                  JSR     R5,SIMBCC               ;GO CALCULATE HEADER CRC
 5555  022546  000020                          16.                             ;16 BITS
 5556  022550  000000                  10$:    .WORD   0                       ;HEADER GOES HERE
 5557  022552  000000                          .WORD   0                       ;START WITH ZERO CRC
 5558  022554  013737  002206  022600          MOV     CALBCC,20$
 5559  022562  013737  002162  022576          MOV     E.MP1,21$
 5560  022570  004537  013130                  JSR     R5,SIMBCC
 5561  022574  000020                          16.
 5562  022576  000000                  21$:    .WORD   0
 5563  022600  000000                  20$:    .WORD   0
 5564  022602  013737  002206  002224          MOV     CALBCC,GDDAT            ;MOVE CALCULATED CRC TO GDDAT
 5565  022610  013737  002164  002226          MOV     E.MP2,BDDAT             ;GET HEADER CRC FROM RLMP
 5566  022616  023737  002224  002226          CMP     GDDAT,BDDAT             ;IS CRC CORRECT?
 5567  022624  001404                          BEQ     11$                     ;IF SO CONTINUE
 5568
 5569  022626                                  ERRDF   53.,EM42,ERR4
   (3) 022626  104462                          TRAP    T$ERCODE
   (5) 022630  000065                          .WORD   53
   (5) 022632  005444                          .WORD   EM42
   (5) 022634  007722                          .WORD   ERR4
 5570  022636                          11$:    CKLOOP                          ;CHECK IF /FL:LOE IS SET
   (3) 022636  104006                          EMT     C$CLP1
 5571
 5572
 5573  022640  005723                          TST     (R3)+                   ;BUMP PATTERN
 5574  022642  020327  002566                  CMP     R3,#SKEND               ;DID WE DO ALL PATTERNS?
 5575  022646  001402                          BEQ     12$                     ;YES, CONTINUE
 5576  022650  000137  022170                  JMP     1$                      ;NO, GO BACK FOR NEXT PATTERN
 5577  022654                          12$:
 5578
```

```
                                ENDSEG                          ;****END OF SEGMENT****
                                100005:
                                        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

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

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

```
                                                CKLOOP                  ;CHECK IF /FL:LOE IS SET
  (3)   022776  104006                          EMT     C$CLP1
5625
5626
5627    023000  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5628    023004  000010                          RDHDR                   ;READ HEADER
5629    023006  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5630    023012                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
  (3)   023012  104006                          EMT     C$CLP1
5631
5632    023014  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5633    023020                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
  (3)   023020  104006                          EMT     C$CLP1
5634
5635    023022  005037  002224                  CLR     GDDAT           ;CLEAR EXPECTED
5636    023026  013737  002226  002240          MOV     BDDAT,DWORD     ;SAVE DIFFERENCE WORD
5637    023034  013737  002160  002226          MOV     E.MP,BDDAT      ;READ HEADER
5638    023042  043737  002176  002226          BIC     SECMSK,BDDAT    ;MASK OUT SECTOR BITS
5639    023050  001404                          BEQ     5$              ;BRANCH IF ON ZERO TRACK
5640
5641    023052                                  ERRDF   54.,EM54,ERR3
  (3)   023052  104462                          TRAP    T$ERCODE
  (5)   023054  000066                          .WORD   54
  (5)   023056  006023                          .WORD   EM54
  (5)   023060  007650                          .WORD   ERR3
5642    023062                          5$:     CKLOOP                  ;CHECK IF /FL:LOE IS SET
  (3)   023062  104006                          EMT     C$CLP1
5643
5644    023064  012777  077601  157032          MOV     #77601,@RLDA    ;GET DIFFERENCE WORD
5645    023072  052777  000005  157024          BIS     #SIGN!MK,@RLDA  ;SET SIGN (TOWARDS SPINDLE) AND MARKER
5646    023100  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5647    023104  000006                          SEEK                    ;SEEK
5648    023106  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5649    023112                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
  (3)   023112  104006                          EMT     C$CLP1
5650
5651
5652    023114  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5653    023120                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
  (3)   023120  104006                          EMT     C$CLP1
5654    023122  012777  000003  156774          MOV     #MK!GSBIT,@RLDA
5655    023130  004537  012552                  JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
5656    023134  000004                          GSTAT
5657    023136  004537  013362                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
5658    023142                                  CKLOOP                  ;CHECK IF /FL:LOE IS SET
  (3)   023142  104006                          EMT     C$CLP1
5659    023144  004537  012252                  JSR     R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
5660
5661    023150                          ENDTST                          ;****END OF TEST****
  (3)   023150                          L10072:
  (3)   023150  104001                          EMT     C$ETST
5662
5663    023152                          BGNMOD  HRDPRM
5664
5665    023152                                  BGNHRD
  (3)   023152  000025                          .WORD L10073-L$HARD/2
```

```
5666
5667  023154                                    GPRML   CNTMSG,CNT,1,YES
 (4)  023154  004130                            .WORD   T$CODE
 (4)  023156  023242                            .WORD   CNTMSG
 (4)  023160  000001                            .WORD   1
5668  023162                                    GPRMA   CSRMSG,CSR,0,160000,177776,YES
 (4)  023162  000031                            .WORD   T$CODE
 (4)  023164  023226                            .WORD   CSRMSG
 (4)  023166  160000                            .WORD   T$LOLIM
 (4)  023170  177776                            .WORD   T$HILIM
5669  023172                                    GPRMA   VECMSG,VECT,0,0,776,YES
 (4)  023172  001031                            .WORD   T$CODE
 (4)  023174  023260                            .WORD   VECMSG
 (4)  023176  000000                            .WORD   T$LOLIM
 (4)  023200  000776                            .WORD   T$HILIM
5670  023202                                    GPRMD   BRMSG,PRIOR,0,340,0,7,YES
 (4)  023202  002032                            .WORD   T$CODE
 (4)  023204  023247                            .WORD   BRMSG
 (4)  023206  000340                            .WORD   340
 (4)  023210  000000                            .WORD   T$LOLIM
 (4)  023212  000007                            .WORD   T$HILIM
5671  023214                                    GPRMD   DRMSG,DRBT,0,03400,0,7,YES
 (4)  023214  003032                            .WORD   T$CODE
 (4)  023216  023267                            .WORD   DRMSG
 (4)  023220  003400                            .WORD   03400
 (4)  023222  000000                            .WORD   T$LOLIM
 (4)  023224  000007                            .WORD   T$HILIM
5672
5673  023226                                    ENDHRD
 (2)                                            .EVEN
 (3)  023226                          L10073:
5674
5675  023226  052502  020123  042101  CSRMSG: .ASCIZ  /BUS ADDRESS/
      023234  051104  051505  000123
5676  023242  046122  030461     000  CNTMSG: .ASCIZ  /RL11/
5677  023247     102  020122  042514  BRMSG:  .ASCIZ  /BR LEVEL/
      023254  042526  000114
5678  023260  042526  052103  051117  VECMSG: .ASCIZ  /VECTOR/
      023266     000
5679  023267     104  044522  042526  DRMSG:  .ASCIZ  /DRIVE/
      023274     000
5680            023276                .EVEN
5681
5682  023276                                    ENDMOD
5683
5684
5685
5686  023276                          BGNMOD  SFTPRM
5687
5688  023276                                    BGNSFT
 (3)  023276  000014                            .WORD L10074-L$SOFT/2
5689  023300                                    GPRML   DMSG,DLT,1,YES
 (4)  023300  000130                            .WORD   T$CODE
 (4)  023302  023330                            .WORD   DMSG
 (4)  023304  000001                            .WORD   1
5690  023306                                    XFERF   I$
```

```
OLTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 88-21
DZRLAA.P11    05-OCT-77 10:41          **TEST 44** - VERIFY GET STATUS WHILE DRDY IS LOW                SEQ 0095

   (5)  023306  006044                          .WORD   T$CODE
 5691   023310                                 GPRMD   EMSG,ELT,0,177777,0,177777,YES
   (4)  023310  001032                          .WORD   T$CODE
   (4)  023312  023365                          .WORD   EMSG
   (4)  023314  177777                          .WORD   177777
   (4)  023316  000000                          .WORD   T$LOLIM
   (4)  023320  177777                          .WORD   T$HILIM
 5692   023322                        1$:       GPRML   SMSG,SIZE,1,YES
   (4)  023322  002130                          .WORD   T$CODE
   (4)  023324  023354                          .WORD   SMSG
   (4)  023326  000001                          .WORD   1
 5693   023330                                 ENDSFT
   (2)                                          .EVEN
   (3)  023330                        L10074:
 5694
 5698
 5699   023330  051104  050117  047440  DMSG:   .ASCIZ  /DROP ON ERROR LIMIT/
 5700   023354  052501  047524  044523  SMSG:   .ASCIZ  /AUTOSIZE/
 5701   023365     105  051122  051117  EMSG:   .ASCIZ  /ERROR LIMIT/
 5702
 5706
 5707           023402                   .EVEN
 5708
 5709   023402                                 ENDMOD
 5710
 5711   023402                        LASTAD
   (2)                                          .EVEN
 5712
```

```
 5713                                   .TITLE  PDP-11 DIAGNOSTIC SUPERVISOR
17009           054302                 END.SUPV=.+2
17011           071776                 .=71776
17012  071776  000000                  .WORD   0
17014           072000                 X1X1=.
17015           000200                 .END 200
```

```
ABOFLA  023654 G        BLD.MW  031320          CSR    = 000000         C$REQT= 000045         EF02  = 000002 G
ABOPAS  023566 G        BLOCK   046454          CSRMSG  023226         C$RESE= 000033         EF03  = 000003 G
ABO.FM  026530          BPRIOR  002132          CSTEST  014026         C$REVI= 000001         EF04  = 000004 G
ADDCOD  012162 G        BRMSG   023247          CURR.T  023602 G       C$RPT = 000025         EF05  = 000005 G
AFREG   004301          BVEC    002134          CYLMSK  002232         C$SEFG= 000047         EF06  = 000006 G
AFTER   013062          B$AAB   032722          C$AAD   036136         C$SPRI= 000041         EF07  = 000007 G
ALLOC   044320          B$AAF   032634          C$AAE   036150         C$SVEC= 000037         EF08  = 000010 G
ARLBA   004236          B.BA    002142          C$AAK   036766         C$TPRI= 000013         EF09  = 000011 G
ARLCS   004231          B.CS    002140          C$AAL   037076         C$UNBU= 000031         EF10  = 000012 G
ARLDA   004244          B.DA    002144          C$ABRT= 000021         C$WTM = 000026         EF11  = 000013 G
ARLMP   004252          B.MP    002146          C$ADR  = 000020         C$WTU = 000027         EF12  = 000014 G
A$AAW   030440          CALBCC  002206          C$AU   = 000054         DAHS  = 000020         EF13  = 000015 G
A$AAX   030454          CALLPC= 000022          C$BRK  = 000022         DATEST  014250         EF14  = 000016 G
A$AAY   030462          CALLPS= 000024          C$BSEG= 000004         DCKMES  003463         EF15  = 000017 G
A$AAZ   030476          CALLSP= 000026          C$BSUB= 000002         DECMSG  042644         EF16  = 000020 G
A$ABA   030506          CALLTC= 000030          C$BUFF= 000030         DEMES   003431         E_    = 000002
BATEST  014146          CAL.CL  051036          C$CEFG= 000046         DERFLG  002150         EMSG    023365
BA16   = 000020         CAL.TI  051074 G        C$CLEA= 000012         DERR   = 040000         EMT.TR  023660 G
BA17   = 000040         CHERR   012252          C$CLP1= 000006         DIAG.T  023662 G       EM1     004375
BCCFBK  002204          CHKLUP  032736          C$CVEC= 000036         DLT    = 000000         EM101   007320
BCSR    002130          CHKSTR  044662          C$DCLN= 000044         DLTMES  003470         EM102   007365
BDDAT   002226          CHKTTY  042750          C$DODU= 000053         DMSG    023330         EM11    004646
BEFORE  013030          CHK.FO  025146          C$DRPT= 000024         DPDVD   053446 G       EM13    004707
BEGPAT  002256          CHK.MA  031076          C$DU   = 000055         DPMUL   053334 G       EM14    004741
BEREG   004260          CHK.PC  036164          C$EDIT= 000006         DRBT   = 000006         EM15    004767
BGN.SU= 023402          CHK.SW  024670          C$ERDF= 000002         DRDY   = 000001         EM16    005015
BINMSG  042630          CHRCNT  044202          C$ERHR= 000003         DRIVE   002136         EM17    005043
BIT0   = 000001 G       CH.FLA  030600          C$ERSF= 000001         DRMSG   023267         EM2     004422
BIT00  = 000001 G       CH.PAS  030622          C$ERSO= 000004         DROP    011114         EM3     004447
BIT01  = 000002 G       CKERLT  012166          C$ESCA= 000010         DRPCOD  012156 G       EM30    005076
BIT02  = 000004 G       CLEAR.  032220          C$ESEG= 000005         DRST   = 000010         EM32    005135
BIT03  = 000010 G       CLKACC  023564 G        C$ESUB= 000003         DRTIM   004347         EM33    005171
BIT04  = 000020 G       CLKBFR  051040          C$ETST= 000001         DSPCOD  011122 G       EM34    005236
BIT05  = 000040 G       CLKCNT  023562 G        C$EXIT= 000032         DS0    = 000000         EM37    005313
BIT06  = 000100 G       CLKRES  052436 G        C$GMAN= 000043         DS1    = 000400         EM4     004474
BIT07  = 000200 G       CLKSER  052736 G        C$GPHR= 000042         DS2    = 001000         EM41    005353
BIT08  = 000400 G       CLKSON  023626 G        C$GPRI= 000040         DS3    = 001400         EM42    005444
BIT09  = 001000 G       CLK.SE  030700          C$GTIM= 000052         DUNIT.  023572 G       EM43    005502
BIT1   = 000002 G       CLNCOD  012110 G        C$INIT= 000011         DVC.FT  036736         EM44    005601
BIT10  = 002000 G       CLR.MA  031154          C$INLP= 000020         DWORD   002240         EM45    005634
BIT11  = 004000 G       CNT    = 000010         C$KWOF= 000035         D$AAG   037606         EM46    005667
BIT12  = 010000 G       CNTMSG  023242          C$KWON= 000034         D$AAH   037624         EM47    005722
BIT13  = 020000 G       CNVT    047114          C$LOOP= 000100         D$AAI   042376         EM5     004521
BIT14  = 040000 G       COMMTA  046734          C$MANI= 000051         D$AAJ   042402         EM52    005753
BIT15  = 100000 G       COMP    003502          C$MSG  = 000023         D$AAK   042420         EM54    006023
BIT2   = 000004 G       CONT    011462          C$PNTB= 000014         D$AAL   042436         EM55    006062
BIT3   = 000010 G       CONTCL  052516 G        C$PNTF= 000017         D$AAM   042446         EM56    006115
BIT4   = 000020 G       CONTIN  011344          C$PNTS= 000016         EF.CON= 000036 G       EM57    006154
BIT5   = 000040 G       CRDY   = 000200         C$PNTX= 000015         EF.NEW= 000035 G       EM6     004572
BIT6   = 000100 G       CRLF    043032          C$POIN= 000040         EF.PWR= 000034 G       EM61    006255
BIT7   = 000200 G       CRTIM   004322          C$QIO  = 000377         EF.RES= 000037 G       EM62    006336
BIT8   = 000400 G       CSEND   002666          C$RDBU= 000007         EF.STA= 000040 G       EM63    006421
BIT9   = 001000 G       CSPAT   002570          C$REFG= 000050         EF01   = 000001 G       EM64    006503
```

```
EM65    006565          FREE    044556          G$OFFS= 000400          I$SUB  = 000041          L$SPC   002030 G
EM66    006646          FRMT1   010432          G$OFSI= 000376          I$TST  = 000041          L$SPCP  002050 G
EM67    006731          FRMT11  010724          G$PRMA= 000001          J$JMP  = 000167          L$SPTP  002060 G
EM7     004620          FRMT12  010765          G$PRMD= 000002          KBPTR   023434 G         L$STA   002066 G
EM70    006766          FRMT13  011044          G$PRML= 000000          KBUF    023436 G         L$SW    011114 G
EM71    007023          FRMT2   010472          G$RADA= 000140          LDCSR   002174           L$TIML  002022 G
EM72    007060          FRMT2A  010511          G$RADB= 000000          LDFUNC  012552           L$TIMU  002020 G
EM73    007113          FRMT2B  010524          G$RADD= 000040          LF      003500           L$TIM1  002016 G
EM74    007146          FRMT3   010553          G$RADF= 000200          LINE.F  023656 G         L$TSTI  002074 G
EM75    007200          FRMT4   010560          G$RADL= 000120          LINE1   010102           L$UNIT  002014 G
EM76    007232          FRMT5   010616          G$RADO= 000020          LINE2   010136           L.CLK.  030166
EM77    007265          FRMT6   010667          G$RADT= 000100          LINE3   010360           L10000  007572
END     012044          FRMT99  010613          G$XFER= 000004          LOAD.F  030620           L10001  007604
ENDPAT  002464          F$AU    = 000015        G$YES = 000010          LOGMSG  042652           L10002  007646
END.OF  032206          F$BGN   = 000040        HALMAX  002544          LPBFR   023432 G         L10003  007720
END.SU= 054302          F$CLEA= 000007          HCORED  030362          LPCNTR  023430 G         L10004  007766
EOP.CH  052760 G        F$DU    = 000016        HCOREQ  030242          LPT.AD  030220           L10005  010000
EOP.FM  026544          F$END   = 000041        HCORET  023614 G        LPT.RE  030214           L10006  010042
EOP.IN  030616          F$HARD= 000004          HCRCME  003450          LSI.RE  030210           L10007  010100
ERR     = 100000        F$HW    = 000013        HDRBUF  002670          LUP     050742           L10010  011112
ERROR   037154          F$INIT= 000006          HDRLST  012772          LUP.AD  036166           L10011  011122
ERRHAN  036170          F$JMP   = 000050        HERTZ.  030202          L$AU    012162 G         L10012  012106
ERRLMT  002246          F$MOD   = 000000        HNFMES  003456          L$AUT   002070 G         L10013  012154
ERRVEC  002202          F$MSG   = 000011        HOLDSP= 000020          L$CCP   002044 G         L10014  012160
ERR.HR  036746          F$PWR   = 000017        HPTCOD  011076 G        L$CLEA  012110 G         L10015  012164
ERR.SF  036752          F$RPT   = 000012        HRDPRM  023152 G        L$DEPO  002011 G         L10016  013314
ERR0    007556 G        F$SEG   = 000003        H$AAB   047442          L$DEVP  002052 G         L10017  013532
ERR1    007574 G        F$SOFT= 000005          ININIT  023604 G        L$DISP  011124 G         L10020  013626
ERRIFO  037240          F$SRV   = 000010        INITCO  011254 G        L$DR    002102 G         L10021  013722
ERR2    007606 G        F$SUB   = 000002        INITIA  042660          L$DRCT  002062 G         L10022  014016
ERR3    007650 G        F$SW    = 000014        INIT.M  031222          L$DRS   002064 G         L10023  014136
ERR4    007722 G        F$TEST= 000001          INIT.R  023416 G        L$DRST  002102 G         L10024  014240
ERR5    007770 G        GARBAG  044204          INPUTA  043606          L$DTP   002040 G         L10025  014326
ERR6    010002 G        GDDAT   002224          INTEN = 000100          L$DU    012156 G         L10026  014452
ERR7    010044 G        GETCHR  042710          INTFLG  002172          L$DUT   002072 G         L10027  014576
ESC.PC  036162          GETCMN  046274          INTFOR  037104          L$DVTY  002104 G         L10030  014702
E.BA    002154          GETPAR  037766          INTSRV  013310          L$EF    002024 G         L10031  015002
E.CS    002152          GETSWI  045270          INVAL.  030330          L$EXP1  002032 G         L10032  015072
E.DA    002156          GET.TW  045040          INVINT  036776          L$EXP2  002034 G         L10033  015172
E.MP    002160          GLBDAT  002112 G        INV.SW  024624          L$EXP3  002036 G         L10034  015302
E.MP1   002162          GLBEQA  002112 G        IN.SUF  032172          L$HARD  023154 G         L10035  015354
E.MP2   002164          GLBERR  007556 G        I$AU   = 000041        L$HPCP  002046 G         L10036  015412
FILL    043500          GLBSUB  012166 G        I$CLN  = 000041        L$HPTP  002056 G         L10037  015536
FILL.C  000204 G        GLBTXT  003370 G        I$DU   = 000041        L$HW    011100 G         L10040  015676
FIRST   002230          GODRVR= 000202          I$HRD  = 000041        L$ICP   002042 G         L10041  016036
FIX     013016          G$BIT  = 000002         I$INIT= 000041         L$INIT  011254 G         L10042  016242
FLAGS   023624 G        G$TAT  = 000004         I$MOD  = 000041        L$LADP  002076 G         L10043  016272
FLAGTA  046652          G$TINT  004172          I$MSG  = 000041        L$LAST  023402 G         L10044  016476
FLAG.I  030664          G$TMES  004133          I$PWR  = 000041        L$MREV  002012 G         L10045  016562
FLA.SE  046620          G$EXCP= 000400          I$RPT  = 000041        L$NAME  002000 G         L10046  016726
FLG.MA  030624          G$HILI= 000002          I$SEG  = 000041        L$REPP  002054 G         L10047  016756
FNDFNC  012770          G$LOLI= 000001          I$SFT  = 000041        L$REV   002010 G         L10050  017044
FORM.T  037250          G$NO   = 000000         I$SRV  = 000041        L$SOFT  023300 G         L10051  017132
```

```
L10052  017260          NUM.LA  037442          RESTMS  012534          SVHD    002244          T$TEST= 000054
L10053  017302          NUM.JN  024000          RE.SET  024772          SWCHAN  030422          T$TSTM= 177777
L10054  017362          NUNITS  032710          RHDINT  004007          SWITCH  047012          T$TSTS= 000001
L10055  017526          NXM     = 020000        RHDMES  003747          SW.PTA  030406          T$$AU = 010015
L10056  017664          NXMMES  003436          RHHS    = 000100        SYS.FT  036726          T$$CLE= 010013
L10057  020202          NXT     011354          RLBA    002122          S$LSYM= 010000          T$$DU = 010014
L10060  020240          NXTFOR  047106          RLCS    002120          TEMP2   002210          T$$HAR= 010073
L10061  020304          OCTMSG  042636          RLDA    002124          TEMP3   002212          T$$HW = 010010
L10062  020430          OKHDR   013002          RLMP    002126          TEMP4   002214          T$$INI= 010012
L10063  021024          OPI     = 002000        RSTACK  053130  G       TERMI   051032          T$$MSG= 010007
L10064  021156          OPIERR  003510          RSX.FL  030614          TERMLI  046640          T$$SEG= 010000
L10065  021320          OPIMES  003443          SEARCH  045006          TERMTA  042622          T$$SOF= 010074
L10066  021460          O$APTS= 000000          SECMSK  002176          TEST.M  030532          T$$SRV= 010016
L10067  021632          O$AU    = 000001        SEEK    = 000006        TIMFLG  023560  G       T$$SW = 010011
L10070  022160          O$BGNR= 000000          SEGSTA  023630  G       TIM.CO  023412  G       T$$TES= 010072
L10071  022656          O$BGNS= 000001          SEKINT  004101          TIM.OP  037246          T.CNTL  002252
L10072  023150          O$DU    = 000001        SEKMES  004050          TMPFNC  002254          T.SIZE  011120
L10073  023226          O$GNSW= 000001          SET.MA  031010          TMP0    002216          T1      013440  G
L10074  023330          O$POIN= 000001          SFTPRM  023276  G       TMP1    002220          T10     014600  G
MAJ.IN  023406  G       PARSES  046346          SHIFT   053766  G       TMP2    002222          T11     014704  G
MAJ.LO  051042          PAR.LA  042340          SIGN    = 000004        TOO.MA  042602          T12     015004  G
MAJ.US  023410  G       PFLG    002166          SIMBCC  013130          TRPFLG  002170          T13     015074  G
MAN.TI  025710          PRINTC  044160          SIZE    = 000004        TRPHAN  013302          T14     015174  G
MAP16   053704  G       PRINTF  047462          SIZE.C  052644  G       TST.AB  033046          T15     015304  G
MASK.B  032734          PRIOR   = 000004        SIZE.M  052562  G       TST.TO  024652          T16     015356  G
MASK.W  032732          PRIO0   = 000000  G     SIZ.TR  052722          TYPEC   043176          T17     015414  G
MAXCYL  002242          PRIO1   = 000040  G     SKEND   002566          TYPEPC  037072          T18     015540  G
MAXSEC  002236          PRIO2   = 000100  G     SKLST   002466          TYPFLA  046514          T19     015700  G
MCHEDR  002000  G       PRIO3   = 000140  G     SMSG    023354          TYPLIN  043074          T2      013534  G
MEM.SI  030230          PRIO4   = 000200  G     SPEC.U  030520          TYPNUM  042462          T20     016040  G
MERLMT  011116          PRIO5   = 000240  G     SPTCOD  011112  G       TYPSTR  043114          T21     016244  G
MIN.IN  023402  G       PRIO6   = 000300  G     SPV.SE  025222          TYP.ER  036756          T22     016274  G
MIN.LS  023404  G       PRIO7   = 000340  G     START   011326          TY.UNI  032212          T23     016500  G
MK      = 000001        PRNTST  044050          STARTC  052512  G       T$ARGC= 000001          T24     016564  G
MODR    053246  G       PRO.CM  030572          STHS    = 000100        T$CODE= 002130          T25     016730  G
MSCRLF  003475          PTAB.S  023612  G       STRCHR  043540          T$ERCO= 000062          T26     016760  G
MUL     053202  G       PUTCHR  042664          STREQ.  030342          T$ERRN= 000066          T27     017046  G
MXSEC1  002234          PWRFLG  002112          STRT.T  030576          T$EXCP= 000000          T28     017134  G
NEWPRI  052726  G       PWR.FA  054140  G       ST.REQ  030516          T$FLAG= 000040          T29     017262  G
NEXTAR  047036          PWR.FL  023414  G       ST.SET  025042          T$HILI= 177777          T3      013630  G
NODRY   003406          PWR.MS  054266          SUNIT.  030604          T$LOLI= 000000          T30     017304  G
NOOPO   = 000000        PWR.SA  054262          SUPERV  026562          T$LSYM= 010000          T31     017364  G
NOOP7   = 000016        PWR.UP  054264          SUPFLA  023570  G       T$MCAL= 177777          T32     017530  G
NOPINT  003614          P.CLK.  030174          SUPV.T  023746  G       T$NEST= 177777          T33     017666  G
NOPMES  003563          QUAMAX  002542          SUP.PR  024614          T$NSK0= 000000          T34     020204  G
NOPWR   011276          RDHDR   = 000010        SVCCNT= 177777          T$NSK1= 000005          T35     020242  G
NORES   003370          READ    = 000014        SVCGBL= 177777          T$SAVL= 177777          T36     020306  G
NC.CLK  025740          READ.P  051044  G       SVCHAN  033114          T$SEGL= 177777          T37     020432  G
NC.FLA  046632          REGBAC  053670  G       SVCINS= 000000          T$SEK0= 010000          T38     021026  G
NC.LPT  044150          REGSAV  053654  G       SVCSTK= 177777          T$SUBN= 000000          T39     021160  G
NC.PTA  030430          REQN.P  030602          SVCSUB= 177777          T$TAGL= 177777          T4      013724  G
NR      = 000000        REQN.T  030574          SVCTAG= 000000          T$TAGN= 010075          T40     021322  G
NUMBIN  037274          REST    011406          SVCTST= 177777          T$TEMP= 000000          T41     021462  G
```

J08

```
DZTERR  MACY11 30(1046)  30-OCT-77  16:51  PAGE 90-3
DZRLAA.SJP    11-OCT-77 15:38          SYMBOL TABLE                                                    SEQ 0100

T42     021634 G        USER.T  023610 G        WRCHK = 000002          XEQ.PR  025750          .XXX  = 072000
T43     022162 G        UUT     002114          WRITE = 000012          XEQ.TE  032472          $BREG   030676
T44     022660 G        VALID.  024050          WTCRDY  013362          XPOLY   002200          $ENDAD  052766 G
T5      014020 G        VAL.LA  024600          WTDRDY  013316          XTIME   051522 G        $SAV2   054032 G
T6      014140 G        VAL.SW  030636          XEQDIA  053014 G        XTIMEN  052346          $SAV3   054046 G
T7      014242 G        VECMSG  023260          XEQSUB  053002 G        XTIMST  051544          $SAV4   054064 G
T8      014330 G        VECT    = 000002        XEQ.CL  032652          XXDP.D  030370          $SAV5   054104 G
T9      014454 G        WCKINT  003706          XEQ.CM  030160          X$ALWA= 000000          .       = 072000
UNITST  002116          WCKMES  003646          XEQ.IN  032534          X$FALS= 000040
UNI.MA  030522          WHY     002250          XEQ.LA  026516          X$OFFS= 000400
USER.P  023606 G        WIDTH   037642          XEQ.OP  032426          X$TRUE= 000020

. ABS.  072000      000

 ERRORS DETECTED:  0

 DSKZ:DZRLAA,DSKZ:DZRLAA=DZRLAA.SML,DZRLAA.P11,DZRLAA.SUP
 RUN-TIME:  36  40  .9 SECONDS
 RUN-TIME RATIO: 342/77=4.3
 CORE USED:  18K  (35 PAGES)
```

K08