

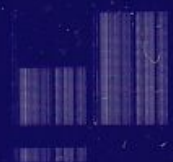
RL11,RLV11

RL11/RLV11 CTLR TST 1 AH-F110B-MC
CZRLGBO FICHE 1 OF 1

MAR 1980
COPYRIGHT © 1977 80
MADE IN USA



Microfilm frame containing a grid of 10 columns and 15 rows of data. The data is extremely faint and illegible due to the low contrast of the scan.



IDENTIFICATION

PRODUCT CODE: AC-F111B-MC
PRODUCT NAME: CZRLGBO RL11/RLV11 CONTROLLER TEST 1
DATE CREATED: 5-JAN-79
REVISED: 7-DEC-79
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHORS: D. DEKNIS, C. CAMPBELL

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

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

COPYRIGHT (C) 1979, DIGITAL EQUIPMENT CORPORATION

TABLE OF CONTENTS

1.0	GENERAL INFORMATION
1.1	PROGRAM ABSTRACT
1.1.1	STRUCTURE OF PROGRAM
1.1.2	DIAGNOSTIC INFORMATION
1.2	SYSTEM REQUIREMENTS
1.2.1	HARDWARE REQUIREMENTS
1.2.2	SOFTWARE REQUIREMENTS
1.3	RELATED DOCUMENTS AND STANDARDS
1.4	DIAGNOSTIC HIERARCHY PREREQUISITES
1.5	ASSUMPTIONS
2.0	OPERATING INSTRUCTIONS
2.1	HOW TO RUN THIS DIAGNOSTIC
2.1.1	THE FIVE STEPS OF EXECUTION
2.1.2	SAMPLE RUN-THROUGH
2.2	CHAIN MODE OPERATION
2.3	DETAILS OF COMMANDS AND SYNTAX
2.3.1	TABLE OF COMMAND VALIDITY
2.3.2	COMMAND SYNTAX
2.4	EXTENDED P-TABLE DIALOGUE
2.5	HARDWARE PARAMETERS
2.6	SOFTWARE PARAMETERS
3.0	ERROR INFORMATION
3.1	ERROR REPORTING
3.2	ERROR HALTS
4.0	PERFORMANCE AND PROGRESS REPORTS
4.1	PERFORMANCE REPORTS
4.2	PROGRESS REPORTS
5.0	DEVICE INFORMATION TABLES
6.0	TEST SUMMARIES

1.0 GENERAL INFORMATION1.1 PROGRAM ABSTRACT1.1.1 STRUCTURE OF PROGRAM

THIS DIAGNOSTIC 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 2.2 'CHAIN MODE OPERATION' FOR DETAILS OF CHAINING PROCEDURE). IT IS A SINGLE PROGRAM FROM THE STANDPOINT OF THE DIAGNOSTIC USER, WHICH AT RUN TIME IS APPENDED TO A COMMON FRONT-END PIECE OF SUPERVISOR SOFTWARE THROUGH WHICH THE DIAGNOSTIC INTERFACES TO THE ENVIRONMENT AS IT EXECUTES.

WHEN THIS DIAGNOSTIC IS STARTED, CONTROL GOES FIRST TO THE SUPERVISOR PORTION, WHICH WILL ASK CERTAIN 'HARD CORE' QUESTIONS ABOUT THE ENVIRONMENT. THEN IT WILL ENTER COMMAND MODE, INDICATED BY A PROMPT CHARACTER (DR>). AT COMMAND MODE THE OPERATOR MAY ENTER ANY OF SEVERAL COMMANDS AS DESCRIBED IN 2.0 'OPERATING INSTRUCTIONS'.

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

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.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

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

1.2.2 SOFTWARE REQUIREMENTS

CZRLGBO RL11/RLV11 CTLR TST 1
(FORMERLY CZRLAB)

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

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

CVRLABO RLV11 RL01 DISKLESS TEST (RLV11 ONLY)

1.5 ASSUMPTIONS

THE HARDWARE OTHER THAN THE RL01/02 SUBSYSTEM IS ASSUMED TO WORK PROPERLY. FALSE ERRORS MAY BE REPORTED IF THE PROCESSOR, ETC., DO NOT FUNCTION PROPERLY.

2.0 OPERATING INSTRUCTIONS

2.1 HOW TO RUN THIS DIAGNOSTIC
-----2.1.1 THE FIVE STEPS OF EXECUTION

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

```
CHMDKAO XXDP+ DK MONITOR NNK
BOOTED VIA UNIT#: 0
ENTER DATE (DD-MM-YY):
```

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

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

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

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

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

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

AT THIS POINT YOU WILL ENTER A 'START' COMMAND. THIS IS NOT THE SAME AS THE XXDP+ 'START' COMMAND, WHICH YOU ALREADY ISSUED IN RESPONSE TO THE XXDP+ DOT PROMPT. THIS 'START' COMMAND CAN TAKE A NUMBER OF SWITCHES AND FLAGS (ALL OPTIONAL) AND THE DETAILS OF THESE ARE SET FORTH IN 2.3 'DETAILS OF COMMANDS AND SYNTAX'. HOWEVER, IN ORDER TO USE THE PROGRAM, ALL YOU NEED TO SAY IS SOMETHING LIKE THIS:

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

THINGS TO NOTE HERE:

1. ONLY THE FIRST THREE CHARACTERS OF THIS OR ANY COMMAND AT THE 'DR>' LEVEL NEED TO BE TYPED.
2. THE 'PASS' SWITCH SPECIFIES HOW MANY PASSES YOU DESIRE. A PASS CONSISTS OF RUNNING THE FULL DIAGNOSTIC AGAINST ALL UNITS BEING TESTED (THIS WILL BE EXPLAINED SHORTLY). ONE PASS IS SPECIFIED IN THE ABOVE EXAMPLE.
3. THE 'FLAGS' SWITCH MAY SPECIFY ANY OF A NUMBER OF FLAGS, BUT THE MAIN USEFUL ONES ARE:

PNT	PRINT NUMBER OF TEST BEING EXECUTED
LOE	LOOP ON ERROR
HOE	HALT ON ERROR
IER	INHIBIT ERROR PRINTOUT

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

* STEP 2 *

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

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

* STEP 3 *

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

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

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

* STEP 4 *

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

* STEP 5 *

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

1. IF NO ERROR IS ENCOUNTERED, THEN THE DIAGNOSTIC WILL SIMPLY EXECUTE THE DESIRED NUMBER OF PASSES AND RETURN TO COMMAND MODE (PROMPT DR>).
2. IF AN ERROR IS ENCOUNTERED, THEN ONE OF THREE THINGS HAPPENS, DEPENDING ON THE SETTINGS OF THE HOE AND LOE FLAGS.

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

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

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

2.1.2 SAMPLE RUN-THROUGH

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

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

1. ISSUE ANOTHER 'START' COMMAND (THUS GOING THRU ALL OF STEPS 1, 2, 3, 4, AND 5 AGAIN).
2. ISSUE A 'RESTART' COMMAND (SAME AS START COMMAND EXCEPT THAT THE HARDWARE QUESTIONS ARE NOT ASKED)
3. ISSUE A 'CONTINUE' COMMAND (EXECUTION WILL RESUME AT THE BEGINNING OF THE PARTICULAR HARDWARE TEST (MOST DIAGNOSTICS CONSIST OF A NUMBER OF THESE) THAT IT WAS IN WHEN THE ERROR HALT OCCURRED. NO QUESTIONS ASKED.
4. ISSUE A 'PROCEED' COMMAND: EXECUTION WILL RESUME AT THE INSTRUCTION FOLLOWING THE ERROR REPORT (THIS IS A SPECIAL COMMAND AND CAN BE ISSUED ONLY AT A HALT ON ERROR).

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

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

THIS WILL DO THE FOLLOWING:

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

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

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

1. START
2. RESTART
3. CONTINUE

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

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

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

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

1. START
2. RESTART

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

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

	BY WHOM ENTERED: -----
.R CZRLGB	O
DRS LOADED	D
DIAG. RUN-TIME SERVICES REV D APR-79	D
CZRLG-B-0	D
CZRLG TESTS CONTROLLER FUNCTIONS, INTERFACE LOGIC, REGISTER OPERATION	D
UNIT IS RL01, RL02	D
DR>STA/PASS:1/FLAGS:HOE	D,O
# UNITS (D) ? 2	D,O
UNIT 0	D
RL11 (L) Y ?	D,O
BUS ADDRESS (O) 174400 ?	D,O
VECTOR (O) 160 ?	D,O
BR LEVEL (O) 5 ?	D,O
DRIVE TYPE = RL01 (L) Y ?	D,O (N=RL02)
DRIVE (O) 0 ?	D,O
UNIT 1	D
RL11 (L) Y ?	D,O
BUS ADDRESS (O) 174400 ?	D,O
VECTOR (O) 160 ?	D,O
BR LEVEL (O) 5 ?	D,O
DRIVE TYPE = RL01 (L) ? Y	D,O (N=RL02)
DRIVE (O) 0 ? 1	D,O
CHANGE SW (L) ? Y	D,O
DROP ON ERROR LIMIT (L) N ?	D,O
CZRLG HRD ERR 00004 TST 003 SUB 002 PC:004130 ERR HLT	
DR>PRO/FLAGS:IER:LOE:HOE-0	D,O

 AT THIS POINT THE DIAGNOSTIC IS LOOPING ON THE
 ERROR WITHOUT PRINTING ANYTHING. YOU CAN SCOPE
 THE ERROR UNTIL YOU HAVE LOCATED IT, THEN ^C OUT.
 TYPING ^C ABORTS THE FUNCTION IN PROGRESS AND
 RETURNS THE XXDP+ MONITOR TO COMMAND MODE.

```
^C                                0
DR>CON/FLAGS:HOE:IER:LOE=0        D,0
CHANGE SW (L) ? N                 D,0
CZRLG EOP 1                        D
^C
DR>RESTART/PASS:1                 D,0
CHANGE SW (L) ? N                 D,0
-----
-----
-----
-----
```

2.2 CHAIN MODE OPERATION

CHAIN MODE OPERATION CONSISTS OF THE SEQUENTIAL EXECUTION OF PROGRAMS WITHOUT OPERATOR INTERVENTION. ONLY PROGRAMS THAT HAVE BEEN MODIFIED TO RUN IN CHAIN MODE CAN BE CHAINED. CHAINABLE PROGRAMS ARE IDENTIFIED IN THE DIRECTORY BY A BIC EXTENSION.

TO RUN CHAIN MODE, THE XXDP+ MONITOR USES AN ASCII FILE (KNOWN AS A CHAIN FILE) LISTING THE PROGRAMS TO BE RUN AND THE NUMBER OF PASSES EACH PROGRAM SHOULD RUN. THIS FILE MUST BE ON THE SYSTEM DEVICE.

A CHAIN FILE MAY BE GENERATED BY USE OF THE XTECO TEXT EDITOR. THE FILE MUST HAVE A CCC EXTENSION. THE CHAIN FILE MAY CONTAIN ANY OF THE COMMANDS SUPPORTED BY THE XXDP+ MONITOR. THE COMMANDS IN THE ASCII FILE ARE EXECUTED IN THE ORDER IN WHICH THEY ARE ENCOUNTERED. COMMENTS MAY BE INCLUDED IN THE FILE.

TO EXECUTE A CHAIN FILE THE USER TYPES:

C FILNAM <CR> OR

C FILNAM/QV<CR>

IN THE FIRST CASE THE PASS COUNT SPECIFIED IN THE CHAIN FILE IS USED BY THE XXDP+ MONITOR TO DETERMINE THE NUMBER OF PASSES TO EXECUTE EACH PROGRAM. IN THE SECOND CASE THE PASS COUNT IS NOT USED AND EACH PROGRAM IS EXECUTED ONLY ONCE. THE /QV SWITCH PROVIDES A SINGLE EXECUTION MODE OF OPERATION OF QUICK VERIFY.

WHEN PROGRAMS ARE RUN IN CHAIN MODE, THE SOFTWARE SWITCH REGISTER SHOULD BE SET TO 000000. THE XXDP+ MONITOR PRINTS EACH COMMAND TAKEN FROM THE CHAIN FILE AND THEN EXECUTES THE COMMAND. WHEN THE LAST COMMAND OTHER THAN ANOTHER C COMMAND HAS BEEN EXECUTED THE XXDP+ MONITOR TERMINATES CHAIN MODE AND TYPES A PROMPT (.). IT IS READY TO ACCEPT ANOTHER COMMAND FROM THE CONSOLE. IF THE LAST COMMAND IS ANOTHER C COMMAND, THE CHAIN MODE WILL CONTINUE AND THE CHAIN FILE SPECIFIED BY THIS NEW C COMMAND WILL BE USED. IF THE USER WISHES TO TERMINATE CHAIN MODE BEFORE ITS NORMAL TERINATION HE MAY DO SO BY TYPING A CNTL-C. HOWEVER, THE MONITOR WILL NOT ABORT THE CHAIN MODE UNTIL IT RECEIVES PROGRAM CONTROL FROM THE PROGRAM CURRENTLY RUNNING.

2.3 DETAILS OF COMMANDS AND SYNTAX

2.3.1 TABLE OF COMMAND VALIDITY

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

<u>HOW ENTERED</u>	<u>LEGAL COMMANDS</u>
1. OPERATOR ENTERED 'RUN DIAG'	START PRINT DISPLAY FLAGS ZFLAGS EXIT
2. DIAGNOSTIC HAS FINISHED ALL ITS REQUESTED PASSES	START RESTART PRINT DISPLAY FLAGS ZFLAGS EXIT
3. OPERATOR INTERRUPTED THE DIAGNOSTIC WITH CTRL/C	START RESTART CONTINUE PRINT DISPLAY FLAGS ZFLAGS EXIT

4. AN ERROR WAS ENCOUNTERED
WITH THE HOE FLAG SET

START
RESTART
CONTINUE
PROCEED
PRINT
DISPLAY
FLAGS
ZFLAGS
EXIT

2.3.2 COMMAND SYNTAX

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

THE DIAGNOSTIC IN CORE IS EXECUTED IN ACCORDANCE WITH THE SWITCHES SPECIFIED. THE MESSAGE '# UNITS?' IS PRINTED. THE START COMMAND MAY BE ISSUED WHEN DIAGNOSTIC COMMAND MODE HAS BEEN ENTERED VIA ONE OF THE FOLLOWING: A) OPERATOR TYPED 'RUN DIAGNOSTIC' B) DIAGNOSTIC FINISHED EXECUTING C) ERROR WAS ENCOUNTERED WITH HOE FLAG SET D) OPERATOR ENTERED CONTROL/C. AFTER THE OPERATOR RESPONDS TO '# UNITS?', THE HARDWARE DIALOGUE IS INITIATED. WHEN IT IS COMPLETED, THE QUESTIONS 'CHANGE SW?' IS ISSUED, AND THE ANSWERS, IF GIVEN, BECOME THE NEW DEFAULTS. THEREFORE IT IS NECESSARY TO RELOAD THE PROGRAM IN ORDER TO RETURN TO THE LOAD DEFAULTS.

THE SWITCH ARGUMENTS ARE AS FOLLOWS:

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

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

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

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

IER INHIBIT ERROR REPORTING
IBE INHIBIT BASIC ERROR REPORTS
IXE INHIBIT EXTENDED ERROR REPORTS
PRI DIRECT ALL MESSAGES TO A LINE PRINTER
PNT PRINT NUMBER OF TEST BEING EXECUTED
BOE BELL ON ERROR
UAM RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS
ISR INHIBIT STATISTICAL REPORTS
IDU INHIBIT DROPPING OF UNITS BY DIAGNOSTIC
ADR EXECUTE AUTODROP CODE
LOT LOOP ON TEST
EVL EVALUATE

THESE FLAGS REPLACE THE USE OF THE HARDWARE SWITCH REGISTER. UNDER THE SUPERVISOR THERE IS NO ACCESS TO THE HARDWARE SWITCH REGISTER.

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

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

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

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

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

THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:

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

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

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

THE SWITCH ARGUMENTS ARE AS IN THE START COMMAND EXCEPT:

1. DEFAULT FOR PASS-CNT IS THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART
2. UNSPECIFIED FLAG SETTINGS ARE UNCHANGED

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

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

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

1. UNSPECIFIED FLAG SETTINGS ARE UNCHANGED

 EXIT

RETURN TO XXDP+ PROMPT MODE.

DRO(P)/UNITS:UNIT-LIST

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

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

ADD/UNITS:UNIT-LIST

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

PRI(NT)

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

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

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

FLA(GS)

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

ZFL(AGS)

ALL FLAGS ARE CLEARED.

2.4 EXTENDED P-TABLE DIALOGUE

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

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

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

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

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

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

_# UNITS (D) ? 8

UNIT 0

RL11 (L) Y ?

BUS ADDRESS (O) 174400 ?

VECTOR (O) 160 ?

BR LEVEL (O) 5 ?

DRIVE TYPE = RL01 (L) Y ?

DRIVE (O) ? 0-3

UNIT 4

RL11 (L) Y ?

BUS ADDRESS (O) 174400 ? 175400

VECTOR (O) 160 ? 164

BR LEVEL (O) 5 ?

DRIVE TYPE = RL01 (L) Y ? N

DRIVE (O) ? 0-3

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

THE SECOND TIME THRU THE P-TABLE QUESTIONS (FOR THE RL02 ASSIGNMENT ON THE SECOND CONTROLLER), THE FIRST QUESTION DEFAULTED TO 'RL11' TYPE CONTROLLER. THE SECOND QUESTION WAS ANSWERED TO REFLECT THE CHANGE IN CSR ADDRESS FOR THE RL02 CONTROLLER (175400). THE SECOND CONTROLLER'S VECTOR WAS ALSO CHANGED TO 164 IN QUESTION #3. THE RL02 TEST UNIT NUMBERS WERE ASSIGNED VALUES 0 TO 3 IN QUESTION #6 AND THE DRIVE TYPE WAS SET FOR RL02'S FOR THE REMAINING 4 UNITS IN QUESTION #5. QUESTION #4 WAS DEFAULTED USING THE 'BR LEVEL' FROM THE FIRST PASS.

2.5 HARDWARE PARAMETERS

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

RL11 (L) Y?

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

BUS ADDRESS (O) 174400?

ANSWER WITH THE BUS ADDRESS OF THE CONTROLLER.

VECTOR (O) 160?

ANSWER WITH THE INTERRUPT VECTOR OF THE CONTROLLER.

BR LEVEL (O) 5?

ANSWER WITH THE INTERRUPT PRIORITY OF THE CONTROLLER.

DRIVE TYPE - RL01 (L) ?

ANSWER NO (N) IF DRIVE IS AN RL02

DRIVE (O) 0?

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

2.6 SOFTWARE PARAMETERS

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

'CHANGE SW ?'

A YES ANSWER WILL ASK THE FOLLOWING SOFTWARE PARAMETER QUESTION, WITH THE PRESENT DEFAULT VALUE PRINTED TO THE LEFT OF THE QUESTION MARK. (THE LAST ANSWER GIVEN IS THE DEFAULT) THE DEFAULT IS TAKEN ON A <CR>.

'DROP ON ERROR LIMIT (L) Y?'

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

3.0 ERROR INFORMATION

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

3.1 ERROR REPORTING

ALL ERROR INFORMATION IS PRINTED ON THE CONSOLE DEVICE. 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 RELEVANT DATA.

EXAMPLE:

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

3.2 ERROR HALTS

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

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS

THIS PROGRAM WILL NOT GIVE ANY PERFORMANCE REPORTS.

4.2 PROGRESS REPORTS

THIS PROGRAM WILL NOT GIVE ANY PROGRESS REPORTS.

5.0 DEVICE INFORMATION TABLES

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

RLCS - CONTROL AND STATUS REGISTER (XXXXX0)

BIT 15 - COMPOSITE ERROR
BIT 14 - DRIVE ERROR
BIT 13 - NON EXISTANT MEMORY ERROR
BIT 12 - HEADER NOT FOUND (WITH BIT 10 SET)
 - DATA LATE (WITH BIT 10 CLEAR)
BIT 11 - HEADER CRC (WITH BIT 10 SET)
 - DATA CRC (WITH BIT 10 CLEAR)
BIT 10 - OPERATION INCOMPLETE
BIT 9/8 - DRIVE SELECT (0-3)
BIT 7 - CONTROLLER READY
BIT 6 - INTERRUPT ENABLE
BIT 5 - EXTENDED BUS ADDRESS (BIT 17)
BIT 4 - EXTENDED BUS ADDRESS (BIT 16)
BIT 3-1 - FUNCTION CODE

- 0 - NOP (PDP-11) MAINT (LSI-11)
- 1 - WRITE CHECK
- 2 - GET DRIVE STATUS
- 3 - SEEK
- 4 - READ HEADER
- 5 - WRITE DATA
- 6 - READ DATA
- 7 - READ WITHOUT HEADER COMPARE

BIT 0 - DRIVE READY

RLBA - BUS ADDRESS REGISTER (XXXXX2)

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

RLDA - DISK ADDRESS REGISTER (XXXXX4)

FOR READ/WRITE FUNCTIONS

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

FOR SEEK FUNCTION

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

FOR GET STATUS FUNCTION

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

RLMP - MULTIPURPOSE REGISTER

FOR READ/WRITE FUNCTION

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

FOR READ HEADER FUNCTION

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

FOR GET STATUS FUNCTION

HAS DRIVE STATUS

BIT 15 - WRITE DATA ERROR
BIT 14 - CURRENT HEAD ERROR (CHE)
BIT 13 - WRITE LOCK STATUS (WL)
BIT 12 - SEEK TIME OUT (SKTO)
BIT 11 - SPIN ERROR (SPE)
BIT 10 - WRITE GATE ERROR (WGE)
BIT 9 - VOLUME CHECK (VC)
BIT 8 - DRIVE SELECT ERROR (DSE)
BIT 7 - DRIVE TYPE IS RLO2 IF SET
BIT 6 - SURFACE (0=UPPPER, 1=LOWER)
BIT 5 - COVER OPEN
BIT 4 - HEADS HOME
BIT 3 - BRUSHES HOME
BIT 2-0 - STATE BITS
 0 - LOAD STATE
 1 - SPIN UP
 2 - BRUSH CYCLE
 3 - LOAD HEADS
 4 - SEEK - TRACK COUNTING
 5 - SEEK - LINEAR MODE
 6 - UNLOAD HEADS
 7 - SPIN DOWN

6.0 TEST SUMMARIES

TEST 1 - RLCS ADDRESSABILITY

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

TEST 2 - RLBA ADDRESSABILITY

THIS TEST WILL CHECK THAT THE BUS ADDRESS REGISTER CAN BE ACCESSED. IF THE REGISTER CANNOT BE ACCESSFD 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 THAT 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 WITH 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 KNOWN DATA, THE RLMP IS WRITTEN, THEN THE RLCS, RLBA AND RLDA ARE VERIFIED THAT THEY DID NOT CHANGE.

TEST 21 - NOOP FUNCTION

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

TEST 22 - TEST NOOP DOES NOTHING

THIS TEST WILL CHECK THAT THE NOOP FUNCTION WILL NOT DISTURB 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 INCOMPLETE) 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 ISSUANCE 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 POSITIONING AT DIFFERENT POINTS ON THE PACK AND CHECKING THAT THE HEADER CRC RECEIVED IS CORRECT.

TEST 44 - VERIFY GET STATUS WHILE DRDY IS LOW

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

@

76	GLOBAL DATA
204	PATTERNS FOR DIFFERENCE WORD
299	GLOBAL TEXT
391	GLOBAL ERRORS
541	LOAD PROTECTION TABLE
548	INITIALIZATION CODE
640	AUTO DROP SECTION
723	GLOBAL SUBROUTINES
745	ROUTINE TO CHECK FOR CONTROLLER ERRORS
825	LOAD RLCS
919	ROUTINE TO CALCULATE CRC
1017	**TEST 1** - RLCS ADDRESSABILITY
1042	**TEST 2** - RLBA ADDRESSABILITY
1068	**TEST 3** - RLDA ADDRESSABILITY
1093	**TEST 4** - RLMP ADDRESSABILITY
1118	**TEST 5** - READ WRITE OF RLCS
1160	**TEST 6** - READ WRITE OF RLBA
1197	**TEST 7** - READ WRITE OF RLDA
1230	**TEST 8** - BIS OF RLCS
1268	**TEST 9** - BIC OF RLCS
1304	**TEST 10** - BIS OF RLBA
1339	**TEST 11** - BIC OF RLBA
1371	**TEST 12** - BIS OF RLDA
1402	**TEST 13** - BIC OF RLDA
1434	**TEST 14** - BUS RESET OF RLCS
1470	**TEST 15** - BUS RESET OF RLBA
1496	**TEST 16** - BUS RESET OF RLDA
1519	**TEST 17** - UNIQUENESS OF RLCS
1561	**TEST 18** - UNIQUENESS OF RLBA
1603	**TEST 19** - UNIQUENESS OF RLDA
1647	**TEST 20** - UNIQUENESS OF RLMP
1700	**TEST 21** - NOOP FUNCTION(RL11 ONLY)
1729	**TEST 22** - TEST NOOP DOES NOTHING
1783	**TEST 23** - TEST OF INTERRUPT
1820	**TEST 24** - TEST PRIORITY BR LEVEL
1871	**TEST 25** - GET STATUS FUNCTION
1896	**TEST 26** - GET STATUS FUNCTION INTERRUPT
1929	**TEST 27** - GET STATUS FUNCTION GENERATES OPI W/O GS BIT
1959	**TEST 28** - OPI UNDER INTERRUPT
1993	**TEST 29** - READ HEADER FUNCTION
2009	**TEST 30** - READ HEADER FUNCTION INTERRUPT
2035	**TEST 31** - REPEATED RD HDRS YIELD SAME CYL AND HD
2083	**TEST 32** - CHECK OF HEADER CRC
2125	**TEST 33** - CHECK CONSECUTIVE HEADERS
2199	**TEST 34** - SEEK FUNCTION
2223	**TEST 35** - CHECK DRIVE READY ON SEEK
2253	**TEST 36** - SEEK FUNCTION INTERRUPT
2299	**TEST 37** - TEST DIFFERENCE WORD TRANSMISSION
2422	**TEST 38** - VERIFY HEAD SELECT 0 VIA RD HDR
2470	**TEST 39** - VERIFY HEAD SELECT 1 VIA RD HDR
2517	**TEST 40** - VERIFY HEAD SELECT 0 VIA GET STATUS
2564	**TEST 41** - VERIFY HEAD SELECT 1 VIA GET STATUS
2612	**TEST 42** - TEST TIME AT WHICH DIF WD GETS TRANSMITTED
2711	**TEST 43** - EXTENSIVE CHECK OF HEADER CRC
2846	**TEST 44** - VERIFY GET STATUS WHILE DRDY IS LOW

1		.TITLE	CZRLGB0 RL11/RLV11 CTLR TST 1
2		.ENABLE	AMA
3		.ENABLE	ABS
4		.NLIST	ME,CND,MD
6			
7	000000	SVC	
8		SVCINS=0	
9		SVCTAG=0	
10		.=2000	
11			
12			
13	002000	POINTER	BGNSFT,BGNSW,BGNDU,BGNAU
14			
15	002000	BGNMOD	MDHEDR
16			
17	002000	HEADER	CZRLG,B,0,4,0
(4)	002000	.ASCII	/C/
(4)	002001	.ASCII	/Z/
(4)	002002	.ASCII	/R/
(4)	002003	.ASCII	/L/
(4)	002004	.ASCII	/G/
(6)	002005	.BYTE	0
(6)	002006	.BYTE	0
(5)	002007	.BYTE	0
(4)	002010	.ASCII	/B/
(4)	002011	.ASCII	/O/
(4)	002012	.WORD	0
(4)	002014	.WORD	4
(4)	002016	.WORD	L\$HARD
(4)	002020	.WORD	L\$SOFT
(4)	002022	.WORD	L\$HW
(4)	002024	.WORD	L\$SW
(4)	002026	.WORD	L\$LAST
(4)	002030	.WORD	0
(4)	002032	.WORD	0
(4)	002034	.WORD	0
(4)	002036	.WORD	0
(4)	002040	.WORD	L\$DISPATCH
(4)	002042	.WORD	0
(4)	002044	.WORD	0
(4)	002046	.WORD	0
(4)	002050	.BYTE	C\$REVISION
(3)	002051	.BYTE	C\$EDIT
(4)	002052	.WORD	0
(5)	002054	.WORD	0
(4)	002056	.WORD	0
(4)	002060	.WORD	L\$DVTYP
(4)	002062	.WORD	0
(4)	002064	.WORD	0
(4)	002066	.WORD	0
(4)	002070	.WORD	L\$AU
(4)	002072	.WORD	L\$DU
(4)	002074	.WORD	0
(4)	002076	.WORD	L\$DESC
(4)	002100	EMT	E\$LOAD
(4)	002102	.WORD	0

(4) 002104 012006 .WORD L\$INIT
(4) 002106 012752 .WORD L\$CLEAN
(4) 002110 012530 .WORD L\$AUTO
(4) 002112 012000 .WORD L\$PROT
(4) 002114 000000 .WORD 0
(4) 002116 000000 .WORD 0
(4) 002120 000000 .WORD 0

18
19 002122 ENDMOD

20
21 002122 DESCRIPT <CZRLG TESTS CONTROLLER FUNCTIONS, INTERFACE LOGIC, REGISTER OPERATION>
(3) 002122 055103 046122 020107 .ASCIZ /CZRLG TESTS CONTROLLER FUNCTIONS, INTERFACE LOGIC, REGISTER OPERATION/
(3) 002130 042524 052123 020123
(3) 002136 047503 052116 047522
(3) 002144 046114 051105 043040
(3) 002152 047125 052103 047511
(3) 002160 051516 020054 047111
(3) 002166 042524 043122 041501
(3) 002174 020105 047514 044507
(3) 002202 026103 051040 043505
(3) 002210 051511 042524 020122
(3) 002216 050117 051105 052101
(3) 002224 047511 000116

(2)
22 002230 DEVTYP .EVEN
(3) 002230 046122 030460 051054 <RL01,RL02>
(3) 002236 030114 000062 .ASCIZ /RL01,RL02/

(2)
23 .EVEN
24 002242 BGNMOD GLBEQAT

25
26 002242 EQUALS

(1) .
(1) . BIT DIFINITIONS
(1) .
(1) 100000 BIT15== 100000
(1) 040000 BIT14== 40000
(1) 020000 BIT13== 20000
(1) 010000 BIT12== 10000
(1) 004000 BIT11== 4000
(1) 002000 BIT10== 2000
(1) 001000 BIT09== 1000
(1) 000400 BIT08== 400
(1) 000200 BIT07== 200
(1) 000100 BIT06== 100
(1) 000040 BIT05== 40
(1) 000020 BIT04== 20
(1) 000010 BIT03== 10
(1) 000004 BIT02== 4
(1) 000002 BIT01== 2
(1) 000001 BIT00== 1
(1) .
(1) 001000 BIT9== BIT09
(1) 000400 BIT8== BIT08
(1) 000200 BIT7== BIT07
(1) 000100 BIT6== BIT06

```
(1) 000040 BIT5== BIT05
(1) 000020 BIT4== BIT04
(1) 000010 BIT3== BIT03
(1) 000004 BIT2== BIT02
(1) 000002 BIT1== BIT01
(1) 000001 BIT0== BIT00
(1) :
(1) : EVENT FLAG DEFINITIONS
(1) : EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
(1) :
(1) 000040 EF.START== 32. : START COMMAND WAS ISSUED
(1) 000037 EF.RESTART== 31. : RESTART COMMAND WAS ISSUED
(1) 000036 EF.CONTINUE== 30. : CONTINUE COMMAND WAS ISSUED
(1) 000035 EF.NEW== 29. : A NEW PASS HAS BEEN STARTED
(1) 000034 EF.PWR== 28. : A POWER-FAIL/POWER-UP OCCURRED
(1) :
(1) : PRIORITY LEVEL DEFINITIONS
(1) :
(1) 000340 PRI07== 340
(1) 000300 PRI06== 300
(1) 000240 PRI05== 240
(1) 000200 PRI04== 200
(1) 000140 PRI03== 140
(1) 000100 PRI02== 100
(1) 000040 PRI01== 40
(1) 000000 PRI00== 0
(1) :
(1) : OPERATOR FLAG BITS
(1) :
(1) 000004 EVL== 4
(1) 000010 LOT== 10
(1) 000020 ADR== 20
(1) 000040 IDU== 40
(1) 000100 ISR== 100
(1) 000200 UAM== 200
(1) 000400 BOE== 400
(1) 001000 PNT== 1000
(1) 002000 PRI== 2000
(1) 004000 IXE== 4000
(1) 010000 IBE== 10000
(1) 020000 IER== 20000
(1) 040000 LOE== 40000
(1) 100000 HOE== 100000
27 000001 DRDY=BIT0 ;DRIVE READY (RLCS)
28 000100 INTEN=BIT6 ;INTERRUPT ENABLE (RLCS)
29 100000 ERR=BIT15 ;RL11 ERROR (RLCS)
30 040000 DERR=BIT14 ;RL01 DRIVE ERROR (RLCS)
31 002000 OPI=BIT10 ;OPERATION INCOMPLETE (RLCS)
32 000200 CRDY=BIT7 ;CONTROLLER READY (RLCS)
33 000040 BA17=BIT5 ;EXTENDED ADDRESS BIT 17 (RLCS)
34 000020 BA16=BIT4 ;EXTENDED ADDRESS BIT 16 (RLCS)
35 020000 NXM=BIT13 ;NON-EXISTANT MEMORY (RLCS)
36 000000 DS0=0 ;DRIVE SELECT 0 (RLCS)
37 000400 DS1=BIT8 ;DRIVE SELECT 1 (RLCS)
38 001000 DS2=BIT9 ;DRIVE SELECT 2 (RLCS)
```

```
39      001400      DS3=BIT8!BIT9      ;DRIVE SELECT 3 (RLCS)
40      000000      NOOP0=0      ;FUNCTION-NOOP(0)
41      000016      NOOP7=BIT1!BIT2.BIT3      ;FUNCTION-NOOP(7)
42      000002      WRCHK=BIT1      ;WRITE CHECK FUNCTION
43      000004      GSTAT=BIT2      ;GET STATUS FUNCTION
44      000006      SEEK=BIT2!BIT1      ;SEEK FUNCTION
45      000010      RDHDR=BIT3      ;READ HEADER FUNCTION
46      000012      WRITE=BIT3!BIT1      ;WRITE DATA FUNCTION
47      000014      READ=BIT3!BIT2      ;READ DATA FUNCTION
48      000202      GODRVR=BIT1!BIT7      ;CRDY AND DRDY
49      000010      DRST=BIT3      ;DRIVE RESET (RLDA)
50      000002      GSBIT=BIT1      ;GET STATUS BIT (RLDA)
51      000001      MK=BIT0      ;MARKER BIT (RLDA)
52      000004      SIGN=BIT2      ;SIGN BIT (RLDA)
53      000100      RHHS=BIT6      ;HEAD SELECT IN READ HEADER
54      000100      STHS=BIT6      ;HEAD SELECT IN STATUS BACK
55      000020      DAHS=BIT4      ;HEAD SELECT IN SEEK
56
57      ;OFFSET FOR HARDWARE P-TABLE
58
59      000000      CSR=0
60      000002      VECT=2
61      000004      PRIOR=4
62      000006      TYPDR=6
63      000010      DRBT=10
64      000012      CNT=12
65
66      ;OFFSET FOR SOFTWARE P-TABLE
67
68      000000      DLT=0
69      000002      ELT=2
70      000004      SIZE=4
71
72 002242      ;      ENDMOD
73
74 002242      BGNMOD  GLBDAT
75
76      .SBTTL  GLOBAL DATA
77
78 002242  000000  PWRFLG: .WORD  0
79 002244  000000  UUT: .WORD  0
80 002246  000000  UNITST: .WORD  0
81 002250  000000  RLCS: .WORD  0
82 002252  000000  RLBA: .WORD  0
83 002254  000000  RLDA: .WORD  0
84 002256  000000  RLMP: .WORD  0
85 002260  000000  BCSR: .WORD  0
86 002262  000000  BPRIOR: .WORD  0
87 002264  000000  BVEC: .WORD  0
88 002266  000000  DRIVE: .WORD  0      ;DRIVE UNDER TEST
89 002270  000000  B.CS: .WORD  0
90 002272  000000  B.BA: .WORD  0
91 002274  000000  B.DA: .WORD  0
92 002276  000000  B.MP: .WORD  0
93 002300  000000  DERFLG: .WORD
94 002302  000000  E.CS: .WORD  0
```

95	002304	000000	E.BA:	.WORD	0	
96	002306	000000	E.DA:	.WORD	0	
97	002310	000000	E.MP:	.WORD	0	
98	002312	000000	E.MP1:	.WORD	0	
99	002314	000000	E.MP2:	.WORD	0	
100	002316	000000	PFLG:	.WORD	0	;PROCESSOR TYPE, 0=UNIBUS, 1=Q-BUS
101	002320	000000	TRPFLG:	.WORD	0	
102	002322	000000	INTFLG:	.WORD	0	;INTERRUPT OCCURRENCE FLAG
103	002324	000000	LDCSR:	.WORD	0	;LOCATION TO FORM RLCS
104	002326	000077	SECMSK:	.WORD	77	;MASK OUT SECTOR
105	002330	120001	XPCLY:	.WORD	120001	
106	002332	000004	ERRVEC:	.WORD	4	
107	002334	000000	BCCFBK:	.WORD	0	;LOCATION USED BY 'SIMBCC'
108	002336	000000	CALBCC:	.WORD	0	;LOCATION USED BY 'SIMBCC'
109	002340	000000	TEMP2:	.WORD	0	;LOCATION USED BY 'SIMBCC'
110	002342	000000	TEMP3:	.WORD	0	;LOCATION USED BY 'SIMBCC'
111	002344	000000	TEMP4:	.WORD	0	;LOCATION USED BY 'SIMBCC'
112	002346	000000	TMP0:	.WORD	0	
113	002350	000000	TMP1:	.WORD	0	
114	002352	000000	TMP2:	.WORD	0	
115	002354	000000	GDDAT:	.WORD	0	
116	002356	000000	BDDAT:	.WORD	0	
117	002360	000000	FIRST:	.WORD	0	;FIRST SECTOR READ
118	002362	177700	CYLMSK:	.WORD	177700	;MASK CYLINDER AND HEAD SELECT
119	002364	000050	MXSEC1:	.WORD	40.	;MAX SECTOR ADDRESS +1
120	002366	000047	MAXSEC:	.WORD	39.	;MAX SECTOR ADDRESS
121	002370	000000	DWORD:	.WORD	0	;DIFFERENCE WORD (SEEK)
122	002372	177600	MAXCYL:	.WORD	177600	;MAXIMUM CYLINDER ADDRESS
123	002374	000000	SVHD:	.WORD	0	;SAVE CURRENT HEAD SELECT
124	002376	000000	WHY:	.WORD	0	;REASON FOR DROP UNIT
125						
126	002400	000000	T.DRIVE:	.WORD	0	
127	002402	000000	T.CNTRLR:	.WORD	0	
128	002404	000000	TMPFNC:	.WORD	0	
129	002406	000000	DLYCNT:	.WORD	0	
130						
131						
132			:			;PATTERNS USED FOR LOADING/READING REGISTERS
133						
134	002410	000000	BEGPAT:	0		;GROWING 1
135	002412	000001		1		
136	002414	000003		3		
137	002416	000007		7		
138	002420	000017		17		
139	002422	000037		37		
140	002424	000077		77		
141	002426	000177		177		
142	002430	000377		377		
143	002432	000777		777		
144	002434	001777		1777		
145	002436	003777		3777		
146	002440	007777		7777		
147	002442	017777		17777		
148	002444	037777		37777		
149	002446	077777		77777		
150	002450	177777		177777		

151	002452	177776	177776	;GROWING 0
152	002454	177774	177774	
153	002456	177770	177770	
154	002460	177760	177760	
155	002462	177740	177740	
156	002464	177700	177700	
157	002466	177600	177600	
158	002470	177400	177400	
159	002472	177000	177000	
160	002474	176000	176000	
161	002476	174000	174000	
162	002500	170000	170000	
163	002502	160000	160000	
164	002504	140000	140000	
165	002506	100000	100000	
166				
167	002510	000000	000000	
168	002512	000001	1	;WALKING 1
169	002514	000002	2	
170	002516	000004	4	
171	002520	000010	10	
172	002522	000020	20	
173	002524	000040	40	
174	002526	000100	100	
175	002530	000200	200	
176	002532	000400	400	
177	002534	001000	1000	
178	002536	002000	2000	
179	002540	004000	4000	
180	002542	010000	10000	
181	002544	020000	20000	
182	002546	040000	40000	
183	002550	100000	100000	
184	002552	177777	177777	;WALKING 0
185	002554	177776	177776	
186	002556	177775	177775	
187	002560	177773	177773	
188	002562	177767	177767	
189	002564	177757	177757	
190	002566	177737	177737	
191	002570	177677	177677	
192	002572	177577	177577	
193	002574	177377	177377	
194	002576	176777	176777	
195	002600	175777	175777	
196	002602	173777	173777	
197	002604	167777	167777	
198	002606	157777	157777	
199	002610	137777	137777	
200	002612	077777	077777	
201	002614	177777	177777	
202	002616	000000	000000	ENDPAT: 000000
203				
204				.SBTTL PATTERNS FOR DIFFERENCE WORD
205				
206	002620	000200		SKLST: .WORD BIT7

207	002622	000400	.WORD	BIT8	;SHIFTING 1
208	002624	001000	.WORD	BIT9	
209	002626	002000	.WORD	BIT10	
210	002630	004000	.WORD	BIT11	
211	002632	010000	.WORD	BIT12	
212	002634	020000	.WORD	BIT13	
213	002636	040000	.WORD	BIT14	
214	002640	077600	.WORD	77600	;SHIFTING 0
215	002642	077400	.WORD	77400	
216	002644	076600	.WORD	76600	
217	002646	075600	.WORD	75600	
218	002650	073600	.WORD	73600	
219	002652	067600	.WORD	67600	
220	002654	057600	.WORD	57600	
221	002656	037600	.WORD	37600	
222	002660	077600	.WORD	77600	
223	002662	000200	.WORD	200	
224	002664	000600	.WORD	600	;GROWING 1
225	002666	001600	.WORD	1600	
226	002670	003600	.WORD	3600	
227	002672	007600	.WORD	7600	
228	002674	017600	QUAMAX: .WORD	17600	
229	002676	037600	HALMAX: .WORD	37600	
230	002700	077600	.WORD	77600	
231	002702	077400	.WORD	77400	;GROWING 0
232	002704	077000	.WORD	77000	
233	002706	076000	.WORD	76000	
234	002710	074000	.WORD	74000	
235	002712	070000	.WORD	70000	
236	002714	060000	.WORD	60000	
237	002716	040000	.WORD	40000	
238	002720	000000	SKEND: .WORD	00000	
239	002722	100000	RL2: .WORD	BIT15	
240	002724	037600	QMAX: .WORD	37600	
241	002726	077600	HMAX: .WORD	77600	
242					
243	002730	177600	.WORD	177600	
244	002732	177400	.WORD	177400	
245	002734	176600	.WORD	176600	
246	002736	173600	.WORD	173600	
247	002740	167600	.WORD	167600	
248	002742	157600	.WORD	157600	
249	002744	137600	.WORD	137600	
250	002746	177000	.WORD	177000	
251	002750	176000	.WORD	176000	
252	002752	174000	.WORD	174000	
253	002754	170000	.WORD	170000	
254	002756	060000	.WORD	60000	
255	002760	040000	.WORD	40000	
256	002762	000000	SKEEND: .WORD	000000	
257					
258					;PATTERNS FOR TEST OF RLCS
259					
260	002764	000000	CSPAT: .WORD	0	;SHIFTING 1
261	002766	000002	.WORD	BIT1	
262	002770	000004	.WORD	BIT2	

263	002772	000010	.WORD	BIT3	
264	002774	000020	.WORD	BIT4	
265	002776	000040	.WORD	BIT5	
266	003000	000100	.WORD	BIT6	
267	003002	000400	.WORD	BIT8	
268	003004	001000	.WORD	BIT9	
269	003006	001576	.WORD	1576	;GROWING 0
270	003010	001574	.WORD	1574	
271	003012	001570	.WORD	1570	
272	003014	001560	.WORD	1560	
273	003016	001540	.WORD	1540	
274	003020	001500	.WORD	1500	
275	003022	001400	.WORD	1400	
276	003024	001576	.WORD	1576	;SHIFT 0
277	003026	001574	.WORD	1574	
278	003030	001566	.WORD	1566	
279	003032	001556	.WORD	1556	
280	003034	001536	.WORD	1536	
281	003036	001436	.WORD	1436	
282	003040	001136	.WORD	1136	
283	003042	000076	.WORD	76	
284	003044	000006	.WORD	6	;GROWING 1
285	003046	000016	.WORD	16	
286	003050	000036	.WORD	36	
287	003052	000076	.WORD	76	
288	003054	000176	.WORD	176	:
289	003056	000576	.WORD	576	
290	003060	001576	.WORD	1576	
291	003062	000000	.WORD	0	
292	003064	000000	ERPOINT: .WORD	0	
293	003066	000100	ERCOUNT: .BLKW	64.	
294	003266	000240	HDRBUF: .BLKW	160.	
295	003766		ENDMOD		
296					

298	003766				BGNMOD	GLBTXT	
299					.SBTTL	GLOBAL	TEXT
300							
304	003766	042040	053122	000	DEMES:	.ASCIZ	/ DRV/
305	003773	040	054116	000115	NXMMES:	.ASCIZ	/ NXM/
306	004000	047440	044520	000	OPIMES:	.ASCIZ	/ OPI/
307	004005	040	041510	041522	HRCMES:	.ASCIZ	/ HCRC/
308	004013	040	047110	000106	HNFMES:	.ASCIZ	/ HNF/
309	004020	042040	045503	000	DCKMES:	.ASCIZ	/ DCK/
310	004025	040	046104	000124	DLTMES:	.ASCIZ	/ DLT/
311	004032	005015	000		MSCRLF:	.ASCIZ	<15><12>
312	004035	015	000		LF:	.ASCIZ	<15>
313	004037	040	047503	050115	COMP:	.ASCIZ	/ COMP/
314	004045	106	051117	042503	OPIERR:	.ASCIZ	/FORCED OPI(GET STATUS) CAUSED OTHER ERRORS/
315	004120	047516	050117	047440	NOPMES:	.ASCIZ	/NOOP OPERATION-FLAG MODE/
316	004151	116	047517	020120	NOPINT:	.ASCIZ	/NOOP OPERATION-INTR. MODE/
317	004203	127	044522	042524	WCKMES:	.ASCIZ	/WRITE CHECK OPERATION-FLAG MODE/
318	004243	127	044522	042524	WCKINT:	.ASCIZ	/WRITE CHECK OPERATION-INTR. MODE/
319	004304	042522	042101	044040	RHDMES:	.ASCIZ	/READ HEADER OPERATION-FLAG MODE/
320	004344	042522	042101	044040	RHDINT:	.ASCIZ	/READ HEADER OPERATION-INTR. MODE/
321	004405	123	042505	020113	SEKMES:	.ASCIZ	/SEEK OPERATION-FLAG MODE/
322	004436	042523	045505	047440	SEKINT:	.ASCIZ	/SEEK OPERATION-INTR. MODE/
323	004470	042507	020124	052123	GSTMES:	.ASCIZ	/GET STATUS OPERATION-FLAG MODE/
324	004527	107	052105	051440	GSTINT:	.ASCIZ	/GET STATUS OPERATION-INTR MODE/
325	004566	051503	020072	000	ARLCS:	.ASCIZ	/CS: /
326	004573	040	040502	020072	ARLBA:	.ASCIZ	/ BA: /
327	004601	040	040504	020072	ARLDA:	.ASCIZ	/ DA: /
328	004607	040	050115	020072	ARLMP:	.ASCIZ	/ MP: /
329	004615	102	043105	051117	BEREG:	.ASCIZ	/BEFORE COMMAND: /
330	004636	044524	042515	047440	AFREG:	.ASCIZ	/TIME OF ERROR: /
331	004657	103	047117	051124	CRTIM:	.ASCIZ	/CONTROLLER TIMED OUT/
332	004704	051104	053111	020105	DRTIM:	.ASCIZ	/DRIVE READY TIMED OUT/
333	004732	040503	020116	047516	EM1:	.ASCIZ	/CAN NOT ADDRESS RLCS/
334	004757	103	047101	047040	EM2:	.ASCIZ	/CAN NOT ADDRESS RLBA/
335	005004	040503	020116	047516	EM3:	.ASCIZ	/CAN NOT ADDRESS RLDA/
336	005031	103	047101	047040	EM4:	.ASCIZ	/CAN NOT ADDRESS RLMP/
337	005056	046122	051503	051040	EM5:	.ASCIZ	%RLCS READ/WRITE ERROR (BIT 0 DON'T CARE)%
338	005127	122	041114	020101	EM6:	.ASCIZ	%RLBA READ/WRITE ERROR%
339	005155	122	042114	020101	EM7:	.ASCIZ	%RLDA READ/WRITE ERROR%
340	005203	117	044520	053440	EM11:	.ASCIZ	/OPI WOULD NOT GENERATE INTERRUPT/
341	005244	047516	044440	052116	EM13:	.ASCIZ	/NO INTERRUPT FROM NOOP(0)/
342	005276	047516	050117	030050	EM14:	.ASCIZ	/NOOP(0) MODIFIED RLMP/
343	005324	047516	050117	030050	EM15:	.ASCIZ	/NOOP(0) MODIFIED RLBA/
344	005352	047516	050117	030050	EM16:	.ASCIZ	/NOOP(0) MODIFIED RLDA/
345	005400	047111	042524	051122	EM17:	.ASCIZ	/INTERRUPT PRIORITY FAILURE/
346	005433	107	052105	051440	EM30:	.ASCIZ	/GET STATUS WOULD NOT INTERRUPT/
347	005472	042507	020124	052123	EM30A:	.ASCIZ	/GET STATUS SHOULD NOT INTERRUPT/
348	005532	046122	050115	041440	EM32:	.ASCIZ	/RLMP CONTAINED WRONG STATUS/
349	005566	050117	020111	044504	EM33:	.ASCIZ	/OPI DID NOT SET-GSTAT WITHOUT GS BIT/
350	005633	117	044520	042040	EM34:	.ASCIZ	/OPI DID NOT SET-GSTAT WITHOUT GS AND MK BITS/
351	005710	042522	042101	044040	EM37:	.ASCIZ	/READ HEADER WOULD NOT INTERRUPT/
352	005750	040502	020104	054503	EM41:	.ASCIZ	/BAD CYLINDER OR HEAD SELECT IN REPEATED READ HEADER TEST/
353	006041	102	042101	044040	EM42:	.ASCIZ	/BAD HEADER CRC ON READ HEADER/
354	006077	123	041505	047524	EM43:	.ASCIZ	/SECTOR ADDRESS OUT OF SEQUENCE DURING CONSECUTIVE READ HEADERS/
355	006176	051127	052111	047111	EM44:	.ASCIZ	/WRITING RLMP MODIFIED RLCS/
356	006231	127	044522	044524	EM45:	.ASCIZ	/WRITING RLMP MODIFIED RLBA/


```
357 006264 051127 052111 047111 EM46: .ASCIZ /WRITING RLMP MODIFIED RLDA/
358 006317 123 042505 020113 EM47: .ASCIZ /SEEK WOULD NOT INTERRUPT/
359 006350 051104 053111 020105 EM52: .ASCIZ /DRIVE READY CAUSED EXTRANEIOUS INTERRUPT/
360 006420 040502 020104 042523 EM54: .ASCIZ /BAD SEEK-TEST OF DIFFENCE WORD/
361 006457 102 042101 044040 EM55: .ASCIZ /BAD HEAD SELECT VIA RD HDR/
362 006512 040502 020104 042510 FM56: .ASCIZ /BAD HEAD SELECT VIA GET STATUS/
363 006551 114 040517 044504 EM57: .ASCII /LOADING RLDA BEFORE DRIVE READY ON SEEK/<15><12>
364 006622 051104 053111 020105 .ASCIZ /DRIVE READY DID NOT SET/
365 006652 044502 020124 042523 EM61: .ASCIZ /BIT SET INSTRUCTION ON RLCS YIELDED WRONG RESULT/
366 006733 102 052111 041440 EM62: .ASCIZ /BIT CLEAR INSTRUCTION ON RLCS YIELDED WRONG RESULT/
367 007016 044502 020124 042523 EM63: .ASCIZ /BIT SET INSTRUCTION ON RLBA YIELDED WRONG RESULT/
368 007077 102 052111 041440 EM64: .ASCIZ /BIT CLEAR INSTRUCTION ON RLBA YIELDED WRONG RESULT/
369 007162 044502 020124 042523 EM65: .ASCIZ /BIT SET INSTRUCTION ON RLDA YIELDED WRONG RESULT/
370 007243 102 052111 041440 EM66: .ASCIZ /BIT CLEAR INSTRUCTION ON RLDA YIELDED WRONG RESULT/
371 007326 052502 020123 042522 EM67: .ASCIZ /BUS RESET DID NOT CLEAR RLCS/
372 007363 102 051525 051040 EM70: .ASCIZ /BUS RESET DID NOT CLEAR RLBA/
373 007420 052502 020123 042522 EM71: .ASCIZ /BUS RESET DID NOT CLEAR RLDA/
374 007455 127 044522 044524 EM72: .ASCIZ /WRITING RLCS MODIFIED RLBA/
375 007510 051127 052111 047111 EM73: .ASCIZ /WRITING RLCS MODIFIED RLDA/
376 007543 127 044522 044524 EM74: .ASCIZ /WRITING RLBA MODIFED RLCS/
377 007575 127 044522 044524 EM75: .ASCIZ /WRITING RLBA MODIFIED RLDA/
378 007627 127 044522 044524 EM76: .ASCIZ /WRITING RLDA MODIFIED RLCS/
379 007662 051127 052111 047111 EM77: .ASCIZ /WRITING RLDA MODIFIED RLBA/
380 007715 122 041514 020123 EM101: .ASCIZ /RLCS CONTAINED FOLLOWING ERROR(S): /
381 007762 000170 EM102: .BLKB 120.
382
383 .EVEN
384
388 010152 ENDMOD
389
```

```
391 .SBTTL GLOBAL ERRORS
392
393 010152 BGNMOD GLBERR
394
395 010152 BGNMSG ERRO
396
397 010152 004737 010476 JSR PC,LINE1
398 010156 004737 010532 JSR PC,LINE2
399
400 010162 004537 013030 JSR R5,CKERLT ;CHECK ERROR LIMIT
401 010166 ENDMSG
(3) 010166 L10000:
(3) 010166 104423 TRAP CSMSG
402
403 010170 BGNMSG ERR1
404
405 010170 004737 010476 JSR PC,LINE1
406
407 010174 004537 013030 JSR R5,CKERLT ;CHECK ERROR LIMIT
408 010200 ENDMSG
(3) 010200 L10001:
(3) 010200 104423 TRAP CSMSG
409
410 010202 BGNMSG ERR2
411
412 010202 004737 010476 JSR PC,LINE1
413 010206 PRINTB #FRMT4,GDDAT,BDDAT
(9) 010206 013746 002356 MOV BDDAT,-(SP)
(8) 010212 013746 002354 MOV GDDAT,-(SP)
(7) 010216 012746 011154 MOV #FRMT4,-(SP)
(6) 010222 012746 000003 MOV #3,-(SP)
(3) 010226 010600 MOV SP,R0
(4) 010230 104414 TRAP CSPNTB
(4) 010232 062706 000010 ADD #10,SP
414
415 010236 004537 013030 JSR R5,CKERLT ;CHECK ERROR LIMIT
416 010242 ENDMSG
(3) 010242 L10002:
(3) 010242 104423 TRAP CSMSG
417
418 010244 BGNMSG ERR3
419
420 010244 004737 010476 JSR PC,LINE1
421 010250 004737 010532 JSR PC,LINE2
422 010254 PRINTB #FRMT5,TMPO,BDDAT,GDDAT
(10) 010254 013746 002354 MOV GDDAT,-(SP)
(9) 010260 013746 002356 MOV BDDAT,-(SP)
(8) 010264 013746 002346 MOV TMPO,-(SP)
(7) 010270 012746 011212 MOV #FRMT5,-(SP)
(6) 010274 012746 000004 MOV #4,-(SP)
(3) 010300 010600 MOV SP,R0
(4) 010302 104414 TRAP CSPNTB
(4) 010304 062706 000012 ADD #12,SP
423
424 010310 004537 013030 JSR R5,CKERLT ;CHECK ERROR LIMIT
425 010314 ENDMSG
```

(3)	010314			L10003:			
(3)	010314	104423			TRAP	C\$MSG	
426							
427	010316			BGNMSG	ERR4		
428							
429	010316	004737	010476		JSR	PC,LINE1	
430	010322	004737	010532		JSR	PC,LINE2	
431	010326				PRINTB	#FRMT4,GDDAT,BDDAT	
(9)	010326	013746	002356		MOV	BDDAT,-(SP)	
(8)	010332	013746	002354		MOV	GDDAT,-(SP)	
(7)	010336	012746	011154		MOV	#FRMT4,-(SP)	
(6)	010342	012746	000003		MOV	#3,-(SP)	
(3)	010346	010600			MOV	SP,R0	
(4)	010350	104414			TRAP	C\$PNTB	
(4)	010352	062706	000010		ADD	#10,SP	
432							
433	010356	004537	013030		JSR	R5,CKERLT	;CHECK ERROR LIMIT
434	010362				ENDMSG		
(3)	010362			L10004:			
(3)	010362	104423			TRAP	C\$MSG	
435							
436	010364			BGNMSG	ERR5		
437							
438	010364	004737	010476		JSR	PC,LINE1	
439							
440	010370	004537	013030		JSR	R5,CKERLT	;CHECK ERROR LIMIT
441	010374				ENDMSG		
(3)	010374			L10005:			
(3)	010374	104423			TRAP	C\$MSG	
442							
443	010376			BGNMSG	ERR6		
444							
445	010376	004737	010476		JSR	PC,LINE1	
446	010402	004737	010754		JSR	PC,LINE3	
447	010406	004737	010532		JSR	PC,LINE2	
448							
449							
450	010412			1\$:	PRINTB	#FRMT99	
(7)	010412	012746	011207		MOV	#FRMT99,-(SP)	
(6)	010416	012746	000001		MOV	#1,-(SP)	
(3)	010422	010600			MOV	SP,R0	
(4)	010424	104414			TRAP	C\$PNTB	
(4)	010426	062706	000004		ADD	#4,SP	
451	010432	004537	013030		JSR	R5,CKERLT	;CHECK ERROR LIMIT
452	010436				ENDMSG		
(3)	010436			L10006:			
(3)	010436	104423			TRAP	C\$MSG	
453							
454	010440			BGNMSG	ERR7		
455							
456	010440	004737	010476		JSR	PC,LINE1	
457	010444				PRINTB	#FRMT6,BDDAT	
(8)	010444	013746	002356		MOV	BDDAT,-(SP)	
(7)	010450	012746	011263		MOV	#FRMT6,-(SP)	
(6)	010454	012746	000002		MOV	#2,-(SP)	
(3)	010460	010600			MOV	SP,R0	

(4)	010462	104414		TRAP	C\$PNTB
(4)	010464	062706	000006	ADD	#6,SP
458					
459	010470	004537	013030	JSR	R5,CKERLT
460					
461	010474			ENDMSG	
(3)	010474			L10007:	
(3)	010474	104423		TRAP	C\$MSG
462					
463	010476			LINE1:	PRINTB #FRMT1,RLCS,<B,DRIVE+1>
(9)	010476	005046		CLR	-(SP)
(9)	010500	153716	002267	BISB	DRIVE+1,(SP)
(8)	010504	013746	002250	MOV	RLCS,-(SP)
(7)	010510	012746	011026	MOV	#FRMT1,-(SP)
(6)	010514	012746	000003	MOV	#3,-(SP)
(3)	010520	010600		MOV	SP,R0
(4)	010522	104414		TRAP	C\$PNTB
(4)	010524	062706	000010	ADD	#10,SP
464	010530	000207		RTS	PC
465					
466	010532			LINE2:	PRINTB #FRMT2,#BEREG,#ARLCS,B.CS,#ARLBA,B.BA
(12)	010532	013746	002272	MOV	B.BA,-(SP)
(11)	010536	012746	004573	MOV	#ARLBA,-(SP)
(10)	010542	013746	002270	MOV	B.CS,-(SP)
(9)	010546	012746	004566	MOV	#ARLCS,-(SP)
(8)	010552	012746	004615	MOV	#BEREG,-(SP)
(7)	010556	012746	011066	MOV	#FRMT2,-(SP)
(6)	010562	012746	000006	MOV	#6,-(SP)
(3)	010566	010600		MOV	SP,R0
(4)	010570	104414		TRAP	C\$PNTB
(4)	010572	062706	000016	ADD	#16,SP
467	010576			PRINTB	#FRMT2A,#ARLDA,B.DA,#ARLMP,B.MP
(11)	010576	013746	002276	MOV	B.MP,-(SP)
(10)	010602	012746	004607	MOV	#ARLMP,-(SP)
(9)	010606	013746	002274	MOV	B.DA,-(SP)
(8)	010612	012746	004601	MOV	#ARLDA,-(SP)
(7)	010616	012746	011105	MOV	#FRMT2A,-(SP)
(6)	010622	012746	000005	MOV	#5,-(SP)
(3)	010626	010600		MOV	SP,R0
(4)	010630	104414		TRAP	C\$PNTB
(4)	010632	062706	000014	ADD	#14,SP
468	010636			PRINTB	#FRMT2,#AFREG,#ARLCS,E.CS,#ARLBA,E.BA
(12)	010636	013746	002304	MOV	E.BA,-(SP)
(11)	010642	012746	004573	MOV	#ARLBA,-(SP)
(10)	010646	013746	002302	MOV	E.CS,-(SP)
(9)	010652	012746	004566	MOV	#ARLCS,-(SP)
(8)	010656	012746	004636	MOV	#AFREG,-(SP)
(7)	010662	012746	011066	MOV	#FRMT2,-(SP)
(6)	010666	012746	000006	MOV	#6,-(SP)
(3)	010672	010600		MOV	SP,R0
(4)	010674	104414		TRAP	C\$PNTB
(4)	010676	062706	000016	ADD	#16,SP
469	010702			PRINTB	#FRMT2B,#ARLDA,E.DA,#ARLMP,E.MP,E.MP1,E.MP2
(13)	010702	013746	002314	MOV	E.MP2,-(SP)
(12)	010706	013746	002312	MOV	E.MP1,-(SP)
(11)	010712	013746	002310	MOV	E.MP,-(SP)

(10) 010716 012746 004607
(9) 010722 013746 002306
(8) 010726 012746 004601
(7) 010732 012746 011120
(6) 010736 012746 000007
(3) 010742 010600
(4) 010744 104414
(4) 010746 062706 000020
470 010752 000207
471
472 010754
(8) 010754 012746 007715
(7) 010760 012746 011147
(6) 010764 012746 000002
(3) 010770 010600
(4) 010772 104414
(4) 010774 062706 000006
473 011000
(8) 011000 012746 007762
(7) 011004 012746 011147
(6) 011010 012746 000002
(3) 011014 010600
(4) 011016 104414
(4) 011020 062706 000006
474 011024 000207
475
479
480 011026 040445 047503 052116
481 011066 047045 052045 052045
482 011105 045 022524 033117
483 011120 052045 047445 022466
484 011147 045 022516 000124
485 011154 047045 040445 054105
486 011207 045 000116
487 011212 047045 040445 040514
488 011263 045 022516 040501
489 011320 040445 051105 047522
490 011361 045 022516 042101
491 011440 047045 052045 040445
492 011471 045 022516 042101
493 011535 045 022516 042101
494
495
496
497
501
502
503
504 011620
505
506 011620
507
508 011620
(3) 011620 000006
509 011622 174400
510 011624 000160

MOV #ARLMP,-(SP)
MOV E.DA,-(SP)
MOV #ARLDA,-(SP)
MOV #FRMT2B,-(SP)
MOV #7,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #20,SP
RTS PC

LINE3: PRINTB #FRMT3,#EM101
MOV #EM101,-(SP)
MOV #FRMT3,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #6,SP
PRINTB #FRMT3,#EM102
MOV #EM102,-(SP)
MOV #FRMT3,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #6,SP
RTS PC

FRMT1: .ASCIZ /%ACONTROLLER: %06%A DRIVE: %01/
FRMT2: .ASCIZ /%N%T%T%06%T%06/
FRMT2A: .ASCIZ /%T%06%T%06/
FRMT2B: .ASCIZ /%T%06%T%06%A %06%A %06/
FRMT3: .ASCIZ /%N%T/
FRMT4: .ASCIZ /%N%AEXP'D: %06%A REC'D: %06/
FRMT99: .ASCIZ /%N/
FRMT5: .ASCIZ /%N%ALAST: %06%A PRES: %06%A EXP'D: %06%N/
FRMT6: .ASCIZ /%N%AAT PROCESSOR LEVEL %06%N/
FRMT11: .ASCIZ /%AERROR LIMIT EXCEEDED-DROPPED%N/
FRMT12: .ASCIZ /%N%ADRIVE DID NOT RECOVER FROM POWER FAILURE%N/
FRMT13: .ASCIZ /%N%T%A - WILL NOT TEST%N/
FRMT14: .ASCIZ /%N%ADRIVE DROPPED - NO CONTROLLER%N/
FRMT15: .ASCIZ /%N%ADRIVE DROPPED - DID NOT RESPOND WITH 'READY'%N/

.EVEN

ENDMOD
BGNMOD HPTCODE
BGNHW
.WORD L10010-L\$HW/2
.WORD 174400 :CSR
.WORD 160 :VECTOR

511	011626	000240			
512	011630	000001	.WORD	240	:PRIORITY
513	011632	000000	.WORD	1	:RL01 = 1
514	011634	000001	.WORD	0	:DRIVE (BITS 8,9,10)
515			.WORD	1	:RL11 = 1, RLV11 = 0
516	011636		ENDHW		
(3)	011636		L10010:		
517					
518	011636		ENDMOD		
519					
520	011636		BGNMOD	SPTCODE	
521					
522	011636		BGNSW		
(3)	011636	000003	.WORD	L10011-L\$SW/2	
523					
524	011640	000000	DROP:	.WORD	0
525	011642	000012	MERLMT:	.WORD	10.
526	011644	000000	T.SIZE:	.WORD	0
527					
528	011646		ENDSW		
(3)	011646		L10011:		
529					
530	011646		ENDMOD		
531					
532	011646		BGNMOD	DSPCODE	
533					
534	011646		DISPATCH	44	
(4)	011646	000054	.WORD	44	
(6)	011650	014346	.WORD	T1	
(6)	011652	014442	.WORD	T2	
(6)	011654	014536	.WORD	T3	
(6)	011656	014632	.WORD	T4	
(6)	011660	014726	.WORD	T5	
(6)	011662	015046	.WORD	T6	
(6)	011664	015150	.WORD	T7	
(6)	011666	015236	.WORD	T8	
(6)	011670	015362	.WORD	T9	
(6)	011672	015506	.WORD	T10	
(6)	011674	015612	.WORD	T11	
(6)	011676	015712	.WORD	T12	
(6)	011700	016002	.WORD	T13	
(6)	011702	016102	.WORD	T14	
(6)	011704	016212	.WORD	T15	
(6)	011706	016264	.WORD	T16	
(6)	011710	016322	.WORD	T17	
(6)	011712	016446	.WORD	T18	
(6)	011714	016606	.WORD	T19	
(6)	011716	016746	.WORD	T20	
(6)	011720	017152	.WORD	T21	
(6)	011722	017202	.WORD	T22	
(6)	011724	017406	.WORD	T23	
(6)	011726	017472	.WORD	T24	
(6)	011730	017636	.WORD	T25	
(6)	011732	017666	.WORD	T26	
(6)	011734	020040	.WORD	T27	
(6)	011736	020126	.WORD	T28	

(6)	011740	020254	.WORD	T29
(6)	011742	020276	.WORD	T30
(6)	011744	020356	.WORD	T31
(6)	011746	020522	.WORD	T32
(6)	011750	020660	.WORD	T33
(6)	011752	021176	.WORD	T34
(6)	011754	021272	.WORD	T35
(6)	011756	021336	.WORD	T36
(6)	011760	021462	.WORD	T37
(6)	011762	022100	.WORD	T38
(6)	011764	022232	.WORD	T39
(6)	011766	022374	.WORD	T40
(6)	011770	022534	.WORD	T41
(6)	011772	022706	.WORD	T42
(6)	011774	023334	.WORD	T43
(6)	011776	024054	.WORD	T44

535
536 012000
537
538

ENDMOD

```
540
541 .SBTTL LOAD PROTECTION TABLE
542 012000 BGNPROT
543 012000 000000 .WORD 0 ;P-TABLE OFFSET OF CSR
544 012002 177777 .WORD -1 ;NOT A MASS-BUS DRIVE
545 012004 000012 .WORD 10. ;P-TABLE OFFSET OF DRIVE
546 012006 ENDPROT
547
548 .SBTTL INITIALIZATION CODE
549 012006 BGNMOD INITCODE
550
551 012006 BGNINIT
552
553 012006 BRESET
(3) 012006 104433 TRAP C$RESET
554 012010 READEF #EF.PWR ;POWER UP?????
(3) 012010 012700 000034 MOV #EF.PWR,R0
(3) 012014 104447 TRAP C$REFG
555 012016 BNCOMPLETE NOPWR ;NO,BRANCH
(2) 012016 103004 BCC NOPWR
556 012020 013737 002012 002242 MOV L$UNIT,PWRFLG ;YES, SET POWER FLAG
557 012026 000475 BR CONT ;GO TO CONTINUE POINT
558 012030 NOPWR: READEF #F.RESTART ;RESTART?
(3) 012030 012700 000037 MOV #EF.RESTART,R0
(3) 012034 104447 TRAP C$REFG
559 012036 BCOMPLETE START1
(2) 012036 103404 BCS START1
560 012040 READEF #EF.START ;START???
(3) 012040 012700 000040 MOV #EF.START,R0
(3) 012044 104447 TRAP C$REFG
561 012046 BNCOMPLETE CONTINUE
(2) 012046 103010 BCC CONTINUE
562 012050 012700 003066 START1: MOV #ERCOUNT,R0
563 012054 012701 000100 MOV #64.,R1
564 012060 005020 1$: CLR (R0)+
565 012062 005301 DEC R1
566 012064 001375 BNE 1$
567 012066 000407 BR START
568
569 012070 CONTINUE: READEF #EF.CONTINUE ;CONTINUE????
(3) 012070 012700 000036 MOV #EF.CONTINUE,R0
(3) 012074 104447 TRAP C$REFG
570 012076 BNCOMPLETE CONT
(2) 012076 103451 BCS CONT
571
572 012100 005737 002244 NXT: TST UUT ;DONE ALL UUT'S
573 012104 001011 BNE XXX ;NO
574 012106 012737 177777 002246 START: MOV #-1,UNITST
575 012114 013737 002012 002244 MOV L$UNIT,UUT
576 012122 012737 003064 003064 MOV #ERCOUNT-2,ERPOINT
577
578 012130 005237 002246 XXX: INC UNITST
579 012134 062737 000002 003064 ADD #2,ERPOINT
580 012142 005337 002244 DEC UUT
581 012146 REST: GPHARD UNITST,R0
(3) 012146 013700 002246 MOV UNITST,R0
```



```

(3) 012414 013700 002246      MOV      UNITST,RO
(3) 012420 104451              TRAP     C$DODU
623 012422                    DOCLN                    ;FORCE AN ABORT
(3) 012422 104444              TRAP     C$DCLN
624 012424 012777 000013 167622 BGNTST: MOV      #13,@RLDA      ;SETUP DR RST
625 012432 012777 000204 167610      MOV      #204,@RLCS      ;GS FUNC
626 012440 053777 002266 167602      BIS      DRIVE,@RLCS      ;SELECT DRIVE
627 012446 042777 000200 167574      BIC      #200,@RLCS      ;ISSUE IT
628 012454 032777 000200 167566 4$:      BIT      #200,@RLCS      ;WAIT FOR READY
629 012462 001774              BEQ      4$
630 012464                    END:      SETVEC   BVEC,#INTSRV,#340
(7) 012464 012746 000340      MOV      #340,-(SP)
(6) 012470 012746 014152      MOV      #INTSRV,-(SP)
(5) 012474 013746 002264      MOV      BVEC,-(SP)
(4) 012500 012746 000003      MOV      #3,-(SP)
(3) 012504 104437              TRAP     C$SVEC
(2) 012506 062706 000010      ADD      #10,SP
631 012512 005037 002316      CLR      PFLG            ;CLR PROCESSOR FLAG
632 012516 104407              READBUS                    ;Q-BUS
(3) 012516 104407              TRAP     C$RDBU
633 012520                    BNCOMPLETE 1$
(2) 012520 103002              BCC      1$
634 012522 005237 002316      INC      PFLG            ;NO, Q-BUS THEN
635 012526                    1$:
636 012526                    ENDINIT
(3) 012526                    L10013:
(3) 012526 104411              TRAP     C$INIT
637
638 012530                    ENDMOD
639
640                    .SBTTL  AUTO DROP SECTION
641 012530                    BGNAUTO
642 012530 005037 002320      CLR      TRPFLG          ;CLEAR TRAP FLAG
643 012534                    SETVEC   ERRVEC,#TRPHAN,#340 ;SET UP VECTOR TO DETECT NON-EXISTENT
(7) 012534 012746 000340      MOV      #340,-(SP)
(6) 012540 012746 014144      MOV      #TRPHAN,-(SP)
(5) 012544 013746 002332      MOV      ERRVEC,-(SP)
(4) 012550 012746 000003      MOV      #3,-(SP)
(3) 012554 104437              TRAP     C$SVEC
(2) 012556 062706 000010      ADD      #10,SP
644
645 012562 012746 000340      MOV      #340,-(SP)
646 012566 012746 014144      MOV      #TRPHAN,-(SP)
647 012572 013746 002332      MOV      ERRVEC,-(SP)
648 012576 012746 000003      MOV      #3,-(SP)
649 012602 104037              EMT      C$SVEC
650 012604 062706 000010      ADD      #10,SP
651 012610 005777 167434      TST      @RLCS          ;ACCESS CONTROLLER
652 012614                    CLRVEC   ERRVEC          ;RELEASE VECTOR
(3) 012614 013700 002332      MOV      ERRVEC,RO
(3) 012620 104436              TRAP     C$CVEC
653 012622 013700 002332      MOV      ERRVEC,RO
654 012626 104036              EMT      C$CVEC
655 012630 005737 002320      TST      TRPFLG          ;DID IT TRAP?
656 012634 001416              BEQ      1$              ;NO - CHECK ITS DRIVE
657 012636                    PRINTB  #FRMT14          ;ELSE, PRINT MSG. 'DRIVE DROPPED - NO CONTROLLER'

```

```
(7) 012636 012746 011471      MOV      #FRMT14,-(SP)
(6) 012642 012746 000001      MOV      #1,-(SP)
(3) 012646 010600              MOV      SP,RO
(4) 012650 104414              TRAP     C$PNTB
(4) 012652 062706 000004      ADD      #4,SP
658 012656 004737 010476      JSR      PC,LINE1          ;PROVIDE DRIVE INFORMATION
659 012662              DODU     UNITST           ;DO DROP UNIT ON DRIVE
(3) 012662 013700 002246      MOV      UNITST,RO
(3) 012666 104451              TRAP     C$DODU
660 012670 000427              BR       2$
661 012672 012777 000200 167350 1$:      MOV      #200,@RLCS        ;EXIT
662 012700 053777 002266 167342      BIS      DRIVE,@RLCS       ;SET CONTROLLER READY
663 012706 032777 000001 167334      BIT      #1,@RLCS         ;SELECT DRIVE
664 012714 001015              BNE      2$                ;IS DRIVE READY?
665 012716              PRINTB  #FRMT15           ;YES - EXIT
(7) 012716 012746 011535      MOV      #FRMT15,-(SP)    ;ELSE, PRINT MSG. 'DRIVE DROPPED - DID NOT
(6) 012722 012746 000001      MOV      #1,-(SP)
(3) 012726 010600              MOV      SP,RO
(4) 012730 104414              TRAP     C$PNTB
(4) 012732 062706 000004      ADD      #4,SP
666              ;/RESPOND WITH 'READY'
667 012736 004737 010476      JSR      PC,LINE1          ;PROVIDE DRIVE INFORMATION
668 012742              DODU     UNITST           ;DO DROP UNIT ON DRIVE
(3) 012742 013700 002246      MOV      UNITST,RO
(3) 012746 104451              TRAP     C$DODU
669 012750              2$:
670 012750              ENDAUTO
(3) 012750              L10014:
(3) 012750 104461              TRAP     C$AUTO
671
672 012752              BGNMOD  CLNCODE
673
674 012752              BGNCLN
675
676 012752              SETPRI  #PRI07
(3) 012752 012700 000340      MOV      #PRI07,RO
(3) 012756 104441              TRAP     C$SPRI
677
678 012760 032777 000200 167262 1$:      BIT      #CRDY,@RLCS
679 012766 001774              BEQ     1$
680
681 012770 042777 000100 167252      BIC      #INTEN,@RLCS
682
683 012776              CLRVEC  BVEC
(3) 012776 013700 002264      MOV      BVEC,RO
(3) 013002 104436              TRAP     C$CVEC
684
685
686
687 013004 005737 002242      TST     PWRFLG            ;TREAT POWER FAILURE
688 013010 001402              BEQ     2$
689
690 013012 005337 002242      DEC     PWRFLG
691
692 013016              2$:
693 013016              ENDCLN
```

(3)	013016		L10015:	
(3)	013016	104412		TRAP C\$CLEAN
694				
695	013020			ENDMOD
696				
697				
698				
699	013020		BGNMOD	DRPCODE
700				
701	013020			BGNDU
702				
703	013020	000240		NOP
704				
705	013022			ENDDU
(3)	013022		L10016:	
(3)	013022	104453		TRAP C\$DU
706				
707	013024			ENDMOD
708				
709	013024		BGNMOD	ADDCODE
710				
711	013024			BGNAU
712				
713	013024	000240		NOP
714				
715	013026			ENDAU
(3)	013026		L10017:	
(3)	013026	104452		TRAP C\$AU
716				
717	013030			ENDMOD
718				
719				
720				

```
722
723      .SBTTL GLOBAL SUBROUTINES
724
725 013030      BGNMOD GLBSUB
726
727
728 013030      CKERLT: INLOOP
(3) 013030 104420      TRAP C$INLP
729 013032      BCOMPLETE 99$
(2) 013032 103427      BCS 99$
730 013034 005737 011640      TST DROP
731 013040 001424      BEQ 99$
732 013042 005277 170016      INC @ERPOINT
733 013046 027737 170012 011642      CMP @ERPOINT, MERLMT
734 013054 002416      BLT 99$
735
736 013056      PRINTF #FRMT11
(7) 013056 012746 011320      MOV #FRMT11, -(SP)
(6) 013062 012746 000001      MOV #1, -(SP)
(3) 013066 010600      MOV SP, R0
(4) 013070 104417      TRAP C$PNTF
(4) 013072 062706 000004      ADD #4, SP
737 013076 004737 010476      JSR PC, LINE1
738 013102      DODU UNITST ;DROP THE UNIT
(3) 013102 013700 002246      MOV UNITST, R0
(3) 013106 104451      TRAP C$DODU
739 013110      DOCLN
(3) 013110 104444      TRAP C$DCLN
740 013112      99$:
741 013112 000205      RTS R5
742
743
744
745      .SBTTL ROUTINE TO CHECK FOR CONTROLLER ERRORS
746
747      ;*****
748      ;*THIS ROUTINE WILL CHECK RLCS FOR ERRORS AND PRINT THEM
749      ;*ACCORDINGLY. IT WILL MERGE THE ERROR PRINTOUT WITH THE TEST
750      ;*ERROR MESSAGE.
751      ;*
752      ;*EXAMPLE: RLCS CONTAINED FOLLOWING ERROR(S):
753      ;*          DRV OPI HCRC HNF
754      ;*          SEEK UNDER INTERRUPT
755      ;*
756      ;*
757      ;*
758      ;*ROUTINE USES R0,R1 AND PICKS HEADER FROM R3
759      ;*
760      ;*      CALL JSR R5,CHERR
761      ;*
762      ;*
763      ;*
764
765 013114 005037 002300      CHERR: CLR DERFLG ;CLEAR OUT DRIVE ERROR FLAG
766 013120 032737 176000 002302      BIT #176000,E.CS ;ANY ERRORS SET
767 013126 001001      BNE 199$ ;IF YES, INVESTIGATE
```

CZRLGB0 RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 1-22
ROUTINE TO CHECK FOR CONTROLLER ERRORS

```

768 013130 000205          RTS      R5          ;NO, EXIT
769 013132 023727 002404 000004 199$:  CMP      TMPFNC,#GSTAT ;FUNCTION-NOP, RESET, GETSTATUS
770 013140 002401          BLT      98$          ;YES, GO CHECK IF ONLY DRIVE ERROR
771 013142 000414          BR       1$          ;YES SERVICE ERROR
772 013144 023727 002404 000002 98$:  CMP      TMPFNC,#WRCHK
773 013152 001410          BEQ      1$          ;
774 013154 013700 002302  MOV      E.CS,R0      ;GET E.CS
775 013160 042700 001777  BIC      #1777,R0      ;
776 013164 022700 140000  CMP      #140000,R0    ;DRIVE ERROR ALONE?
777 013170 001001          BNE      1$          ;NO, GO SERVICE
778 013172 000205          RTS      R5          ;YES, EXIT
779
780 013174 012701 007762 1$:  MOV      #EM102,R1    ;GET START OF STRING
781 013200 005737 002302  TST      E.CS          ;IS COMPOSITE ERROR SET?(BETTER BE)
782 013204 100003          BPL      99$          ;IT'S NOT SOMETHING IS WRONG
783 013206 004537 013660  JSR      R5,FIX        ;YES, PUT 'COMP' IN STRING
784 013212 004037          COMP
785 013214 032737 040000 002302 99$:  BIT      #DERR,E.CS    ;DRIVE ERROR SET?
786 013222 001405          BEQ      3$          ;NO, CONTINUE
787 013224 005237 002300  INC      DERFLG        ;SET DRV ERROR FLAG
788 013230 004537 013660  JSR      R5,FIX        ;YES, PUT 'DRV' INTO STRING
789 013234 003766          DEMES
790 013236 032737 020000 002302 3$:  BIT      #NXM,E.CS     ;NON-EXISTENT MEMORY ERROR?
791 013244 001403          BEQ      4$          ;NO, CONTINUE
792 013246 004537 013660  JSR      R5,FIX        ;YES, PUT 'NXM' INTO STRING
793 013252 003773          NXMMES
794 013254 032737 002000 002302 4$:  BIT      #OPI,E.CS     ;IS OPI SET?
795 013262 001422          BEQ      6$          ;NO, GO CHECK BITS 11 & 12
796 013264 004537 013660  JSR      R5,FIX        ;PUT 'OPI' INTO STRING
797 013270 004000          OPI:MES
798 013272 032737 004000 002302  BIT      #BIT11,E.CS    ;HEADERCRC ERROR?
799 013300 001403          BEQ      5$          ;NO, GO CHECK HEADER NOT FOUND
800 013302 004537 013660  JSR      R5,FIX        ;GO PUT 'HCRC' IN STRING
801 013306 004005          HCRCMES
802 013310 032737 010000 002302 5$:  BIT      #BIT12,E.CS    ;HEADER NOT FOUND?
803 013316 001422          BEQ      8$          ;NO, GO PUT 'CRLF' IN STRING
804 013320 004537 013660  JSR      R5,FIX        ;PUT 'HNF' IN STRING
805 013324 004013          HNF:MES
806 013326 000416          BR       8$          ;PUT 'CRLF' IN STRING
807 013330 032737 004000 002302 6$:  BIT      #BIT11,E.CS    ;DATA CRC ERROR?
808 013336 001403          BEQ      7$          ;NO, GO CHECK DATA LATE
809 013340 004537 013660  JSR      R5,FIX        ;PUT 'DCK' IN STRING
810 013344 004020          DCKMES
811 013346 032737 010000 002302 7$:  BIT      #BIT12,E.CS    ;DATA LATE ERROR?
812 013354 001403          BEQ      8$          ;NO, GO PUT IN 'CRLF'
813 013356 004537 013660  JSR      R5,FIX        ;PUT 'DLT' IN STRING
814 013362 004025          DLT:MES
815 013364 004537 013660 8$:  JSR      R5,FIX
816 013370 004032          MSCRLF
817 013372 004537 013660  JSR      R5,FIX
818 013376 000000          RESTMS: .WORD      0          ;HEADER FROM TEST
819 013400 105011          CLR     (R1)         ;PUT TERMINATOR IN
820
821 013402          ERRDF      300.,LF,ERR6
(4) 013402 104455          TRAP      C$ERDF
(5) 013404 000454          .WORD      300

```

```

(5) 013406 004035          .WORD  LF
(5) 013410 010376          .WORD  ERR6
822
823 013412 000205          RTS    R5          ;EXIT ROUTINE
824
825          .SBTTL  LOAD RLCS
826          :*****
827          :* ROUTINE TO LOAD RLCS WITH FUNCTION TO BE PERFORMED
828          :*      CALL:   JSR    R5,LDFUNC
829          :*          .WORD          ;BITS TO BE LOADED, FUNCTION
830          :*          ;AND INTR ENABLE ONLY
831          :*
832          :
833
834 013414 012537 002324      LDFUNC: MOV    (R5)+,LDCSR      ;GET BITS TO LOAD
835 013420 005737 002300      TST    DERFLG
836 013424 001424          BEQ    98$
837 013426 013746 002270      MOV    B.CS,-(SP)
838 013432 012777 000013 166614  MOV    #13,@RLDA
839 013440 012737 000004 002270  MOV    #GSTAT,B.CS
840 013446 053737 002266 002270  BIS    DRIVE,B.CS
841 013454 013777 002270 166566  MOV    B.CS,@RLCS
842 013462 012637 002270      MOV    (SP)+,B.CS
843 013466 032777 000200 166554 99$:  BIT    #200,@RLCS
844 013474 001774          BEQ    99$
845 013476 010346          98$:  MOV    R3,-(SP)      ;SAVE R3
846 013500 042737 177661 002324  BIC    #177661,LDCSR  ;CLEAR ALL BUT FUNC & INTR EN
847 013506 013737 002324 013632  MOV    LDCSR,FNDFNC  ;SAVE FUNCTION
848 013514 042737 000100 013632  BIC    #INTEN,FNDFNC ;ONLY FUNCTION
849 013522 013737 013632 002404  MOV    FNDFNC,TMPFNC
850 013530 012703 013634      MOV    #HDRLST,R3    ;GET HEADER LIST
851 013534 006237 013632      ASR    FNDFNC        ;ALIGN TO RIGHT
852 013540 001404          BEQ    2$
853 013542 022323          1$:  CMP    (R3)+,(R3)+   ;BUMP R3 BY 4
854 013544 005337 013632      DEC    FNDFNC        ;FOUND IT
855 013550 001374          BNE    1$           ;NO,KEEP LOOKING
856 013552 032737 000100 002324 2$:  BIT    #INTEN,LDCSR  ;YES,DO WE WANT FLAG OR INTR
857 013560 001401          BEQ    3$           ;FLAG BRANCH
858 013562 005723          TST    (R3)+        ;INTR POINT TO THAT ONE
859 013564 011303          3$:  MOV    (R3),R3      ;SET HEADER
860 013566 010337 013376      MOV    R3,RESTMS    ;SET UP HEADER
861 013572 053737 002266 002324  BIS    DRIVE,LDCSR  ;SELECT DRIVE
862 013600 052737 000200 002324 4$:  BIS    #200,LDCSR   ;CONTROLLER READY
863 013606 013777 002324 166434  MOV    LDCSR,@RLCS
864 013614 004537 013672      JSR    R5,BEFORE
865 013620 042777 000200 166422 5$:  BIC    #200,@RLCS
866 013626 012603          MOV    (SP)+,R3    ;RESTORE R3
867 013630 000205          RTS    R5          ;EXIT
868
869 013632 000000          FNDFNC: .WORD  0
870
871 013634 004120          HDRLST: NOPMES
872 013636 004151          NOINT
873 013640 004203          WCKMES
874 013642 004243          WCKINT
875 013644 004470          OKHDR:  GSTMES

```

```
876 013646 004527          GSTINT
877 013650 004405          SEKMES
878 013652 004436          SEKINT
879 013654 004304          RHDMES
880 013656 004344          RHDINT
881
882
883      ;*****
884      ;*ROUTINE TO MOVE ASCII STRINGS
885      ;*USES REGISTERS R1 - WHERE STRING IS BEING BUILT
886      ;*
887      ;*      CALL      JSR      R5, FIX
888      ;*      .WORD      .WORD      ;ADDRESS OF STRING TO MOVE
889
889 013660 012500      FIX:  MOV      (R5)+, R0      ;GET ADDRESS AND MOVE RETURN
890 013662 112021      1$:  MOVB     (R0)+, (R1)+    ;GET BYTE AND UPDATE
891 013664 001376          BNE      1$      ;WATCH 0 BYTE TERMINATOR
892 013666 105741          TSTB     -(R1)     ;BACK UP OVER ZERO BYTE
893 013670 000205          RTS      R5      ;EXIT
894
895
896      ;LOAD REGISTERS BEFORE FUNCTION
897      ;CALL:  JSR      R5, BEFORE
898
899 013672 017737 166352 002270  BEFORE:  MOV      @RLCS, B.CS      ;READ CS
900 013700 017737 166346 002272          MOV      @RLBA, B.BA      ;READ BA
901 013706 017737 166342 002274          MOV      @RLDA, B.DA      ;READ DA
902 013714 017737 166336 002276          MOV      @RLMP, B.MP      ;READ MP
903 013722 000205          RTS      R5
904
905
906      ;LOAD REGISTERS AT ERROR
907      ;CALL:  JSR      R5, AFTER
908
909 013724 017737 166320 002302  AFTER:  MOV      @RLCS, E.CS      ;READ CS
910 013732 017737 166314 002304          MOV      @RLBA, E.BA      ;READ BA
911 013740 017737 166310 002306          MOV      @RLDA, E.DA      ;READ DA
912 013746 017737 166304 002310          MOV      @RLMP, E.MP      ;READ MP
913 013754 017737 166276 002312          MOV      @RLMP, E.MP1     ;READ MP
914 013762 017737 166270 002314          MOV      @RLMP, E.MP2     ;READ MP
915 013770 000205          RTS      R5
916
917
918
919      .SBTTL  ROUTINE TO CALCULATE CRC
920
921      ;ROUTINE WILL CALCULATE A CRC-16 CRC ON A WORD OF
922      ;1-16 BITS IN LENGTH, RESULT IS RETURNED IN 'CALBCC'
923      ;
924      ;
925      ;      CALL:  JSR      R5, SIMBCC
926      ;      .WORD      .WORD      ;NUMBER OF BITS (1-16)
927      ;      .WORD      .WORD      ;DATA FOR CRC CALCULATION
928      ;      .WORD      .WORD      ;PREVIOUS OR STARTING CRC
929      ;      ;      (SHOULD BE ZEROED FOR START)
930      ;
931      ;      ROUTINE USES R0, R1, R2
931 013772 010046      SIMBCC:  MOV      R0, -(SP)      ;SAVE R0
```



```
932 013774 010146      MOV      R1,-(SP)      ;SAVE R1
933 013776 010246      MOV      R2,-(SP)      ;SAVE R2
934 014000 012537 002340  MOV      (R5)+,TEMP2    ;GET NUMBER OF BITS
935 014004 012537 002342  MOV      (R5)+,TEMP3    ;GET DATA FOR CRC CALCULATION
936 014010 012537 002344  MOV      (R5)+,TEMP4    ;GET STARTING CRC
937 014014 005037 002334  1$: CLR      BCCFBK      ;
938 014020 013700 002344  MOV      TEMP4,R0      ;GET PRESENT CRC
939 014024 006037 002342  ROR      TEMP3          ;ROTATE NEW DATA
940 014030 005500      ADC      R0             ;MERGE NEW WITH OLD
941 014032 032700 000001  BIT      #1,R0          ;BIT 0 SET
942 014036 001402      BEQ      2$            ;IF NOT CONTINUE
943 014040 005137 002334  COM      BCCFBK        ;
944 014044 013700 002330  2$: MOV      XPOLY,R0    ;GET CRC POLYNOMIAL (CRC-16)
945 014050 005100      COM      R0            ;COMPLIMENT POLYNOMIAL
946 014052 040037 002334  BIC      R0,BCCFBK     ;
947 014056 000241      CLC                      ;CLEAR CARRY
948 014060 006037 002344  ROR      TEMP4          ;
949 014064 013700 002334  MOV      BCCFBK,R0     ;
950 014070 013701 002344  MOV      TEMP4,R1     ;
951 014074 010102      MOV      R1,R2         ;
952 014076 040100      BIC      R1,R0         ;
953 014100 043702 002334  BIC      BCCFBK,R2     ;
954 014104 050200      BIS      R2,R0         ;
955 014106 043737 002330 002344  BIC      XPOLY,TEMP4   ;
956 014114 050037 002344  BIS      R0,TEMP4     ;
957 014120 005337 002340  DEC      TEMP2         ;
958 014124 001333      BNE      1$            ;
959 014126 013737 002344 002336  MOV      TEMP4,CALBCC  ;
960 014134 012602      MOV      (SP)+,R2     ;
961 014136 012601      MOV      (SP)+,R1     ;
962 014140 012600      MOV      (SP)+,R0     ;
963 014142 000205      RTS      R5            ;RETURN
```

```
964  
965  
966  
967 ;ROUTINE TO SET FLAG IF TRAP OCCURRED  
968 ;'TRPHAN' IS IN LOCATION 4.
```

```
969  
970  
971 014144 005237 002320  TRPHAN: INC      TRPFLG      ;INDICATE TRAP  
972 014150 000002      RTI                      ;RETURN
```

```
973  
974 014152      BGNSRV
```

```
975  
976 014152 005237 002322  INTSRV: INC      INTFLG      ;INDICATE INTERRUPT  
977
```

```
978 014156      ENDSRV  
(3) 014156  
(2) 014156 000002      L10020: RTI
```

```
979  
980 ;ROUTINE TO WAIT FOR DRIVE READY
```

```
981 014160 010146      WTD RDY: MOV      R1,-(SP)    ;SAVE R1  
982 014162 012701 003720  MOV      #200.,R1      ;TIME OUT OF 200 MILLISECONDS  
983 014166 032777 000001  *66054 1$: BIT      #DRDY,@RLCS    ;DRIVE READY?  
984 014174 001022      BNE      2$            ;YES, EXIT  
985 014176      DELAY   #1            ;WAIT A WHILE
```

```

(2) 014176 012727 000001      MOV    ##1,(PC)+
(2) 014202 000000      .WORD 0
(2) 014204 013727 002116      MOV    L$DLY,(PC)+
(2) 014210 000000      .WORD 0
(2) 014212 005367 177772      DEC    -6(PC)
(2) 014216 001375      BNE    -.4
(2) 014220 005367 177756      DEC    -22(PC)
(2) 014224 001367      BNE    -.20
986 014226 005301      DEC    R1          ;CHECK IF TIME UP
987 014230 001356      BNE    1$          ;NO, GO CHECK DRIVE READY
988
989 014232      ERRDF 200.,DRTIM,ERR5 ;DRIVE READY DID NOT SET
(4) 014232 104455      TRAP  C$ERDF
(5) 014234 000310      .WORD 200
(5) 014236 004704      .WORD DRTIM
(5) 014240 010364      .WORD ERR5
990
991 014242 012601      2$:  MOV    (SP)+,R1      ;RESTORE
992 014244 000205      RTS    R5              ;EXIT
993
994      ;ROUTINE TO WAIT FOR CONTROLLER READY
995 014246 010146      WTCRDY: MOV    R1,-(SP)      ;SAVE R1
996 014250 012701 017500      MOV    #8000.,R1      ;WAIT 800 MILLISECONDS
997 014254 032777 000200 165766 1$: BIT    #CRDY,@RLCS      ;CONTROLLER READY
998 014262 001025      BNE    2$              ;YES, EXIT
999 014264      DELAY #1              ;WAIT A WHILE
(2) 014264 012727 000001      MOV    ##1,(PC)+
(2) 014270 000000      .WORD 0
(2) 014272 013727 002116      MOV    L$DLY,(PC)+
(2) 014276 000000      .WORD 0
(2) 014300 005367 177772      DEC    -6(PC)
(2) 014304 001375      BNE    -.4
(2) 014306 005367 177756      DEC    -22(PC)
(2) 014312 001367      BNE    -.20
1000 014314 005301      DEC    R1          ;CHECK IF TIME UP
1001 014316 001356      BNE    1$          ;NO GO BACK
1002
1003 014320 004537 013724      JSR    R5,AFTER      ;GET REGISTERS
1004
1005 014324      ERRDF 100.,CRTIM,ERR6 ;CONTROLLER TIMED OUT
(4) 014324 104455      TRAP  C$ERDF
(5) 014326 000144      .WORD 100
(5) 014330 004657      .WORD CRTIM
(5) 014332 010376      .WORD ERR6
1006
1007 014334 000402      BR     3$              ;EXIT
1008
1009 014336 004537 013724      2$:  JSR    R5,AFTER      ;GET REGISTERS
1010 014342 012601      3$:  MOV    (SP)+,R1
1011 014344 000205      RTS    R5              ;EXIT
1012
1013 014346      ENDMOD
1014
1015

```

```
1017 .SBTTL **TEST 1** - RLCS ADDRESSABILITY
1018
1019 014346 BGNSTST ;****START OF TEST****
1020 014346 STARS
(2) ;*****
1021 ;TEST TO SEE IF WE CAN ADDRESS THE CONTROL
1022 ;AND STATUS REGISTER. IF WE TRAP WE WILL REPORT
1023 ;THE ERROR AND ABORT. AFTER THIS TEST WE ONLY KNOW
1024 ;THAT WE CAN ADDRESS THE REGISTER.
1025 014346 STARS
(2) ;*****
1026
1027
1028 014346 005037 002320 1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
1029 014352 2$: SETVEC ERRVEC,#TRPHAN,#340 ;SET TO CATCH TRAP
(7) 014352 012746 000340 MOV #340,-(SP)
(6) 014356 012746 014144 MOV #TRPHAN,-(SP)
(5) 014362 013746 002332 MOV ERRVEC,-(SP)
(4) 014366 012746 000003 MOV #3,-(SP)
(3) 014372 104437 TRAP C$SVEC
(2) 014374 062706 000010 ADD #10,SP
1030
1031 014400 005777 165644 TST @RLCS ;ADDRESS RLCS
1032 014404 CLRVEC ERRVEC ;RELEASE TRAP VECTOR
(3) 014404 013700 002332 MOV ERRVEC,R0
(3) 014410 104436 TRAP C$CVEC
1033 014412 005737 002320 TST TRPFLG ;TRAP OCCURRED???
1034 014416 001407 BEQ 3$ ;NO, IKAY PROCEED
1035 014420 013737 002250 002354 MOV RLCS,GDDAT ;SET UP ERROR DATA
1036
1037 014426 ERRSF 0.,EM1,ERR1 ;BUS TIMEOUT IN ADDRESSING RLCS
(4) 014426 104454 TRAP C$ERSF
(5) 014430 000000 .WORD 0
(5) 014432 004732 .WORD EM1
(5) 014434 010170 .WORD ERR1
1038 014436 3$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 014436 104406 TRAP C$CLP1
1039 014440 ENDTST ;****END OF TEST****
(3) 014440 L10021:
(3) 014440 104401 TRAP C$ETST
1040
1041 .SBTTL **TEST 2** - RLBA ADDRESSABILITY
1042
1043 BGNSTST ;****START OF TEST****
1044 014442
1045 STARS
1046 ;*****
1047 014442 ;TEST TO SEE IF WE CAN ADDRESS THE BUS ADDRESS
(2) ;REGISTER. IF WE TRAP WE WILL REPORT THE ERROR
1048 ;AND ABORT. AFTER THIS TEST WE ONLY KNOW THAT
1049 ;WE CAN ADDRESS THE REGISTER.
1050 STARS
1051 ;*****
1052 014442
(2)
1053
```

```
1054 014442 005037 002320 1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
1055 014446 012746 000340 2$: SETVEC ERRVEC,#TRPHAN,#340 ;SET TO CATCH TRAP
(7) 014446 012746 000340 MOV #340,-(SP)
(6) 014452 012746 014144 MOV #TRPHAN,-(SP)
(5) 014456 013746 002332 MOV ERRVEC,-(SP)
(4) 014462 012746 000003 MOV #3,-(SP)
(3) 014466 104437 TRAP C$SVEC
(2) 014470 062706 000010 ADD #10,SP
1056
1057 014474 005777 165552 TST @RLBA ;ADDRESS RLBA
1058 014500 CLRVEC ERRVEC ;RELEASE TRAP VECTOR
(3) 014500 013700 002332 MOV ERRVEC,R0
(3) 014504 104436 TRAP C$CVEC
1059 014506 005737 002320 TST TRPFLG ;TRAP OCCURRED???
1060 014512 001407 BEQ 3$ ;NO, CONTINUE
1061 014514 013737 002252 002354 MOV RLBA,GDDAT ;SETUP ERROR DATA
1062
1063 014522 ERRSF 1,EM2,ERR1 ;BUS TIMEOUT IN ADDRESSING RLBA
(4) 014522 104454 TRAP C$ERSF
(5) 014524 000001 .WORD 1
(5) 014526 004757 .WORD EM2
(5) 014530 010170 .WORD ERR1
1064 014532 3$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 014532 104406 TRAP C$CLP1
1065 014534 ENDTST ;****END OF TEST****
(3) 014534 L10022:
(3) 014534 104401 TRAP C$ETST
1066
1067
1068
1069 .SBTTL **TEST 3** - RLDA ADDRESSABILITY
1070 014536 BGNST ;****START OF TEST****
1071 014536 STARS
(2) ;*****
1072 ;TEST TO SEE IF WE CAN ADDRESS THE DISK ADDRESS
1073 ;REGISTER IF WE TRAP WE WILL REPORT THE ERROR
1074 ;AND ABORT. AFTER THIS TEST WE ONLY KNOW THAT
1075 ;WE CAN ADDRESS THE REGISTER.
1076 014536 STARS
(2) ;*****
1077
1078
1079 014536 005037 002320 1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
1080 014542 012746 000340 2$: SETVEC ERRVEC,#TRPHAN,#340 ;SET TO CATCH TRAP
(7) 014542 012746 000340 MOV #340,-(SP)
(6) 014546 012746 014144 MOV #TRPHAN,-(SP)
(5) 014552 013746 002332 MOV ERRVEC,-(SP)
(4) 014556 012746 000003 MOV #3,-(SP)
(3) 014562 104437 TRAP C$SVEC
(2) 014564 062706 000010 ADD #10,SP
1081
1082 014570 005777 165460 TST @RLDA ;ADDRESS RLDA
1083 014574 CLRVEC ERRVEC ;RELEASE TRAP VECTOR
(3) 014574 013700 002332 MOV ERRVEC,R0
(3) 014600 104436 TRAP C$CVEC
1084 014602 005737 002320 TST TRPFLG ;TRAP OCCURRED???
```

```

1085 014606 001407          BEQ      3$          ;NO, CONTINUE
1086
1087 014610 013737 002254 002354  MOV     RLDA,GDDAT    ;SETUP ERROR INFO
1088 014616          ERRSF   2.,EM3,ERR1 ;BUS TIMEOUT IN ADDRESSING RLDA
(4) 014616 104454          TRAP   C$ERSF
(5) 014620 000002          .WORD  2
(5) 014622 005004          .WORD  EM3
(5) 014624 010170          .WORD  ERR1
1089 014626          3$:      CKLOOP          ;CHECK IF /FL.LOE IS SET
(3) 014626 104406          TRAP   C$CLP1
1090 014630          ENDTST          ;****END OF TEST****
(3) 014630          L10023:
(3) 014630 104401          TRAP   C$ETST
1091
1092
1093
1094

```

.SBTTL **TEST 4** - RLMP ADDRESSABILITY

```

1095 014632          BGNTST          ;****START OF TEST****
1096 014632          STARS
(2)
1097          ;:*****
1098          ;TEST TO SEE IF WE CAN ADDRESS THE MULTIPURPOSE
1099          ;REGISTER. IF WE TRAP WE WILL REPORT THE ERROR AND
1100          ;ABURT. AFTER THIS TEST WE ONLY KNOW THAT WE CAN
1101          ;ADDRESS THE REGISTER.
1101 014632          STARS
(2)
1102          ;:*****
1103

```

```

1104 014632 005037 002320          1$:      CLR      TRPFLG          ;CLEAR TRAP OCCURANCE
1105 014636          2$:      SETVEC   ERRVEC,#TRPHAN,#340 ;SET UP TO CATCH TRAP
(7) 014636 012746 000340          MOV     #340,-(SP)
(6) 014642 012746 014144          MOV     #TRPHAN,-(SP)
(5) 014646 013746 002332          MOV     ERRVEC,-(SP)
(4) 014652 012746 000003          MOV     #3,-(SP)
(3) 014656 104437          TRAP   C$SVEC
(2) 014660 062706 000010          ADD     #10,SP
1106
1107 014664 005777 165366          TST     @RLMP          ;ADDRESS RLMP
1108 014670          CLRVEC   ERRVEC          ;RELEASE TRAP VECTOR
(3) 014670 013700 002332          MOV     ERRVEC,RO
(3) 014674 104436          TRAP   C$CVEC
1109 014676 005737 002320          TST     TRPFLG          ;TRAP OCCURRED???
1110 014702 001407          BEQ     3$          ;NO, CONTINUE
1111 014704 013737 002256 002354  MOV     RLMP,GDDAT    ;SET UP ERROR INFO
1112
1113 014712          ERRSF   3.,EM4,ERR1 ;BUS TIMEOUT IN ADDRESSING RLMP
(4) 014712 104454          TRAP   C$ERSF
(5) 014714 000003          .WORD  3
(5) 014716 005031          .WORD  EM4
(5) 014720 010170          .WORD  ERR1
1114 014722          3$:      CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 014722 104406          TRAP   C$CLP1
1115 014724          ENDTST          ;****END OF TEST****
(3) 014724          L10024:
(3) 014724 104401          TRAP   C$ETST
1116

```

```
1117
1118 .SBTTL **TEST 5** - READ WRITE OF RLCS
1119
1120 014726 BGNTST ;****START OF TEST****
1121
1122
1123
1124 014726 STARS
(2) ;:*****
1125 ;:TEST THAT WE CAN WRITE/READ BITS 8,9 AND BITS 6-1
1126 ;:OF THE CONTROL AND STATUS REGISTER. BITS 15-10 AND 0
1127 ;:ARE DON'T CARE BITS AT THIS TIME AND BIT 7
1128 ;:(CONTROLLER READY) IS ALWAYS WRITTEN TO A ONE.
1129 014726 STARS
(2) ;:*****
1130
1131
1132 014726 012703 002764 MOV #CSPAT,R3 ;SET UP TABLE POINTER OF PATTERNS
1133
1134 014732 BGNSEG ;****START OF SEGMENT****
(3) 014732 104404 TRAP C$BSEG
1135
1136 014734 C$TEST:
1137 014734 011337 002354 MOV (R3),GDDAT ;GET PATTERN INTO GDDAT
1138 014740 052737 000200 002354 BIS #200,GDDAT ;INSURE GO IS SET
1139 014746 013777 002354 165274 MOV GDDAT,@RLCS ;LOAD RLCS (CONTROL AND STATUS)
1140 014754 032777 040000 165266 BIT #DERR,@RLCS ;IF DRIVE ERROR PRESENT
1141 014762 001403 BEQ 99$ ;THEN EXPECT DRIVE AND
1142 014764 052737 140000 002354 BIS #ERR!DERR,GDDAT ;COMPOSITE ERROR
1143 014772 017737 165252 002356 99$: MOV @RLCS,BDDAT ;READ RLCS BACK
1144 015000 042737 000001 002356 BIC #DRDY,BDDAT ;IGNORE DRIVE READY
1145 015006 023737 002354 002356 CMP GDDAT,BDDAT ;DID WE READ WHAT WE LOADED
1146 015014 001404 BEQ 1$ ;YES, THEN BRANCH
1147
1148 015016 ERRDF 4,EM5,ERR2 ;WRONG DATA IN RLCS
(4) 015016 104455 TRAP C$ERRDF
(5) 015020 000004 .WORD 4
(5) 015022 005056 .WORD EM5
(5) 015024 010202 .WORD ERR2
1149 015026 1$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 015026 104410 TRAP C$ESCAPE
(3) 015030 000012 .WORD 10000$-.
1150
1151
1152 015032 005723 TST (R3)+ ;BUMP FOR NEXT PATTERN
1153 015034 020327 003062 CMP R3,#CSEND ;CHECK FOR END
1154 015040 001335 BNE C$TEST ;NOT END, LOAD NEXT PATTERN
1155
1156 015042 ENDSEG ;****END OF SEGMENT****
(3) 015042 10000$: TRAP C$ESEG
(3) 015042 104405
1157 015044 ENDTST ;****END OF TEST****
(3) 015044 L10025: TRAP C$ETST
(3) 015044 104401
1158
1159
```

```
1160 .SBTTL **TEST 6** - READ WRITE OF RLBA
1161
1162 015046 BGNTST ;****START OF TEST****
1163
1164 015046 STARS
(2) ;*****
1165 ;TEST THAT WE CAN WRITE/READ BITS 15-1 OF THE
1166 ;BUS ADDRESS REGISTER. FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
1167 ;GROWING 0 AND SHIFTING 0. BIT 0 IS ALSO LOADED BUT
1168 ;SHOULD ALWAYS COME BACK AS 0
1169 015046 STARS
(2) ;*****
1170
1171
1172 015046 012703 002410 BGNSEG MOV #BEGPAT,R3 ;GET START OF PATTERN LIST
1173 015052 TRAP C$BSEG ;****START OF SEGMENT****
(3) 015052 104404
1174 015054 BATEST:
1175 015054 011337 002354 MOV (R3),GDDAT ;GET PATTERN TO SEND
1176 015060 005737 002402 TST T.CNTRL ;RL11??
1177 015064 001403 BEQ 2$ ;NO
1178 015066 042737 000001 002354 BIC #BIT0,GDDAT ;KEEP RLBA EVEN (UNIBUS)
1179 015074 013777 002354 165150 2$: MOV GDDAT,@RLBA ;LOAD PATTERN TO BUS ADDRESS
1180 015102 017737 165144 002356 MOV @RLBA,BDDAT ;READ IT BACK
1181 015110 023737 002354 002356 CMP GDDAT,BDDAT ;IS IT CORRECT?
1182 015116 001404 BEQ 1$ ;IF SO, BRANCH
1183
1184 015120 ERRDF 5,EM6,ERR2 ;DATA WRONG IN RLBA
(4) 015120 104455 TRAP C$ERRDF
(5) 015122 000005 .WORD 5
(5) 015124 005127 .WORD EM6
(5) 015126 010202 .WORD ERR2
1185 015130 1$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 015130 104410 TRAP C$ESCAPE
(3) 015132 000012 .WORD 10000$-
```

```

1187
1188
1189 015134 005723          TST      (R3)+          ;BUMP FOR NEXT PATTERN
1190 015136 020327 002616  CMP      R3,#ENDPAT    ;CHECK FOR END
1191 015142 001344          BNE      BATEST        ;NOT END, BRANCH FOR NEXT
1192
1193 015144          ENDSEG          ;****END OF SEGMENT****
(3) 015144          10000$:
(3) 015144 104405          TRAP     C$ESEG
1194 015146          ENDTST          ;****END OF TEST****
(3) 015146          L10026:
(3) 015146 104401          TRAP     C$ETST
1195
1196
1197          .SBTTL  **TEST 7** - READ WRITE OF RLDA
1198
1199 015150          BGNST          ;****START OF TEST****
1200
1201 015150          STARS
(2)          ;:*****
1202          ;:TEST THAT WE CAN WRITE/READ THE DISK ADDRESS REGISTER
1203          ;:ALL BIT POSITIONS ARE WRITTEN USING FOUR PATTERNS:
1204          ;:GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
1205 015150          STARS
(2)          ;:*****
1206
1207
1208 015150 012703 002410          MOV      #BEGPAT,R3    ;SET UP POINTER TO PATTERN LIST
1209 015154          BGNSEG          ;****START OF SEGMENT****
(3) 015154 104404          TRAP     C$BSEG
1210 015156          DATEST:
1211 015156 011337 002354          MOV      (R3),GDDAT    ;GET PATTERN
1212 015162 013777 002354 165064  MOV      GDDAT,@RLDA   ;LOAD PATTERN IN DA
1213
1214 015170 017737 165060 002356          MOV      @RLDA,BDDAT   ;READ PATTERN BACK
1215 015176 023737 002354 002356          CMP      GDDA*,BDDAT   ;IS IT CORRECT?
1216 015204 001404          BEQ      1$           ;BRANCH IF CORRECT
1217
1218 015206          ERRDF 6.,EM7,ERR2  ;WRONG DATA IN RLDA
(4) 015206 104455          TRAP     C$ERDF
(5) 015210 000006          .WORD   6
(5) 015212 005155          .WORD   EM7
(5) 015214 010202          .WORD   ERR2
1219 015216          1$: ESCAPE SEG          ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 015216 104410          TRAP     C$ESCAPE
(3) 015220 000012          .WORD   10000$-.
1220
1221
1222 015222 005723          TST      (R3)+          ;BUMP POINTER
1223 015224 020327 002616  CMP      R3,#ENDPAT    ;AT END OF PATTERNS?
1224 015230 001352          BNE      DATEST        ;NO, BRANCH BACK
1225
1226 015232          ENDSEG          ;****END OF SEGMENT****
(3) 015232          10000$:
(3) 015232 104405          TRAP     C$ESEG
1227 015234          ENDTST          ;****END OF TEST****

```



```

(3) 015234 L10027: TRAP C$ETST
(3) 015234 104401
1228
1229
1230 .SBTTL **TEST 8** - BIS OF RLCS
1231
1232 015236 BGNTST ;****START OF TEST****
1233 015236 STARS
(2) ;*****
1234 ;TEST THAT WE CAN USE THE 'BIS' INSTRUCTION ON THE CONTROL
1235 ;AND STATUS REGISTER. BITS 8,9 AND 6-1 ARE TESTED TO
1236 ;SET INDIVIDUALLY AS WELL AS COLLECTIVELY WITHOUT DESTROYING
1237 ;ANY PREVIOUS DATA PATTERN
1238 015236 STARS
(2) ;*****
1239
1240
1241 015236 012703 002764 BGNSEG MOV #CSPAT,R3 ;GET BEGINNING OF LIST
1242 015242 TRAP C$BSEG ;****START OF SEGMENT****
(3) 015242 104404
1243 015244 1$:
1244 015244 012777 000200 164776 MOV #CRDY,@RLCS ;INSURE GO IS THERE
1245 015252 011337 002354 MOV (R3),GDDAT ;SET UP EXPECTED RLCS
1246 015256 052737 000200 002354 BIS #CRDY,GDDAT ;IN GDDAT
1247 015264 051377 164760 BIS (R3),@RLCS ;BIT SET PATTERN IN RLCS
1248 015270 032777 040000 164752 BIT #DERR,@RLCS ;IF ERROR BIT SET THEN
1249 015276 001403 BEQ 99$ ;EXPECT IT ON THE READ
1250 015300 052737 140000 002354 BIS #ERR!DERR,GDDAT ;BACK
1251 015306 017737 164736 002356 99$: MOV @RLCS,BDDAT ;READ RLCS TO CHECK 'BIS'
1252 015314 042737 000001 002356 BIC #DRDY,BDDAT ;CLEAR OUT DRIVE READY
1253 015322 023737 002356 002354 CMP BDDAT,GDDAT ;DID BIS WORK?
1254 015330 001404 BEQ 2$ ;BRANCH IF OKAY
1255
1256 015332 ERRDF 7.,EM61,ERR2 ;WRONG DATA IN RLCS
(4) 015332 104455 TRAP C$ERDF
(5) 015334 000007 .WORD 7
(5) 015336 006652 .WORD EM61
(5) 015340 010202 .WORD ERR2
1257 015342 2$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 015342 104410 TRAP C$ESCAPE
(3) 015344 000012 .WORD 10000$-.
1258 ;BIT OR CLEARED OTHER BIT
1259
1260 015346 005723 TST (R3)+ ;GET NEXT PATTERN
1261 015350 022703 003062 CMP #CSEND,R3 ;AT END OF LIST
1262 015354 001333 BNE 1$ ;NG GO BACK FOR TEST OF
1263 ;NEXT PATTERN
1264 015356 ENDSEG ;****END OF SEGMENT****
(3) 015356 10000$:
(3) 015356 104405 TRAP C$ESEG
1265 015360 ENDTST ;****END OF TEST****
(3) 015360 L10030:
(3) 015360 104401 TRAP C$ETST
1266
1267
1268 .SBTTL **TEST 9** - BIC OF RLCS
  
```

```
1269
1270 015362          BGNTST          ;****START OF TEST****
1271
1272 015362          STARS
(2)                ;*****
1273                ;TEST THAT THE 'BIC' INSTRUCTION WILL WORK ON THE
1274                ;CONTROL AND STATUS REGISTER. BITS 8-9 AND 6-1 ARE
1275                ;TESTED.
1276 015362          STARS
(2)                ;*****
1277
1278
1279 015362 012703 002764      MOV      #CSPAT,R3      ;GET BEGINNING OF PATTERNS
1280 015366          BGNSEG          ;****START OF SEGMENT****
(3) 015366 104404          TRAP      C$BSEG
1281 015370          1$:
1282 015370 012777 001776 164652      MOV      #1776,@RLCS    ;SET ALL SETTABLE BITS
1283 015376 012737 001776 002354      MOV      #1776,GDDAT    ;SET UP EXPECT DATA IN
1284 015404 041337 002354          BIC      (R3),GDDAT      ;GDDAT
1285 015410 041377 164634          BIC      (R3),@RLCS     ;CLEAR BITS IN RLCS VIA 'BIC'
1286 015414 032777 040000 164626      BIT      #DERR,@RLCS   ;IF DRIVE ERROR BIT SET
1287 015422 001403          BEQ      99$            ;EXPECT IT SET WHEN WE
1288 015424 052737 140000 002354      BIS      #ERR!DERR,GDDAT ;READ IT BACK
1289 015432 017737 164612 002356      99$:  MOV      @RLCS,BDDAT    ;MOVE RLCS TO BDDAT FOR COMPARE
1290 015440 042737 000001 002356      BIC      #DRDY,BDDAT    ;CLEAR DRIVE READY
1291 015446 023737 002356 002356      CMP      BDDAT,GDDAT    ;DID 'BIC' WORK PROPERLY
1292 015454 001404          BEQ      2$            ;BRANCH IF OKAY
1293
1294 015456          ERRDF      8.,EM62,ERR2 ;WRONG DATA IN RLCS
(4) 015456 104455          TRAP      C$ERDF
(5) 015460 000010          .WORD      8
(5) 015462 006733          .WORD      EM62
(5) 015464 010202          .WORD      ERR2
1295 015466          2$:  ESCAPE  SEG          ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 015466 104410          TRAP      C$ESCAPE
(3) 015470 000012          .WORD      10000$-.
1296
1297 015472 005723          TST      (R3)+          ;GET NEXT PATTERN
1298 015474 020327 003062      CMP      R3,#CSEND      ;AT END OF LIST
1299 015500 001333          BNE      1$            ;NO, GO BACK WITH NEXT PATTERN
1300 015502          ENDSEG          ;****END OF SEGMENT****
(3) 015502 10000$:
(3) 015502 104405          TRAP      C$ESEG
1301 015504          ENDTST          ;****END OF TEST****
(3) 015504  L10031:
(3) 015504 104401          TRAP      C$ETST
1302
1303
1304          .SBTTL  **TEST 10** - BIS OF RLBA
1305
1306 015506          BGNTST          ;****START OF TEST****
1307
1308 015506          STARS
(2)                ;*****
1309                ;TEST THAT THE 'BIS' INSTRUCTION WILL WORK ON THE BUS
1310                ;ADDRESS REGISTER. BITS 15-0 ARE LOADED, ONLY BITS 15-1
```

```
1311 ;ARE EXPECTED BACK. FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,  
1312 ;GROWING 0, AND SHIFTING 0.  
1313 015506 STARS  
(2) ;*****  
1314  
1315  
1316 015506 012703 002410 BGNSEG MOV #BEGPAT,R3 ;GET START OF LIST  
1317 015512 TRAP C$BSEG ;****START OF SEGMENT****  
(3) 015512 104404  
1318 015514 1$: CLR @RLBA ;CLEAR 'BA'  
1319 015514 005077 164532 MOV (R3),GDDAT ;SET EXPECTED  
1320 015520 011337 002354 TST T.CNTRL ;RL11  
1321 015524 005737 002402 BEQ 3$ ;NO  
1322 015530 001403 BIC #1,GDDAT ;BIT 0 CAN'T SET IN RLBA (UNIBUS)  
1323 015532 042737 000001 002354 3$: BIS (R3),@RLBA ;BIS RLBA WITH PATTERN  
1324 015540 051377 164506 MOV @RLBA,BDDAT ;READ 'BA'  
1325 015544 017737 164502 002356 CMP BDDAT,GDDAT ;DID RLBA LOAD PROPERLY?  
1326 015552 023737 002356 002354 BEQ 2$ ;BRANCH IF YES  
1327 015560 001404  
1328  
1329 015562 ERRDF 9,EM63,ERR2 ;WRONG DATA IN RLBA  
(4) 015562 104455 TRAP C$ERRDF  
(5) 015564 000011 .WORD 9  
(5) 015566 007016 .WORD EM63  
(5) 015570 010202 .WORD ERR2  
1330 015572 2$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG  
(3) 015572 104410 TRAP C$ESCAPE  
(3) 015574 000012 .WORD 10000$-.  
1331  
1332 015576 005723 TST (R3)+ ;GET NEXT PATTERN  
1333 015600 020327 002616 CMP R3,#ENDPAT ;DID WE COMPLETE LIST  
1334 015604 001343 BNE 1$ ;NO, GO BACK FOR NEXT.  
1335 015606 ENDSEG ;****END OF SEGMENT****  
(3) 015606 10000$: TRAP C$ESEG  
(3) 015606 104405  
1336 015610 ENDTST ;****END OF TEST****  
(3) 015610 L10032: TRAP C$ETST  
(3) 015610 104401  
1337  
1338  
1339 .SBTTL **TEST 11** - BIC OF RLBA  
1340  
1341 015612 BGNST ;****START OF TEST****  
1342  
1343 015612 STARS  
(2) ;*****  
1344 ;TEST THAT THE 'BIC' INSTRUCTION WILL WORK ON THE BUS  
1345 ;ADDRESS REGISTER. BITS 15-1 ARE TESTED WITH 4 PATTERNS  
1346 ;GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0.  
1347 015612 STARS  
(2) ;*****  
1348  
1349  
1350 015612 012703 002410 BGNSEG MOV #BEGPAT,R3 ;GET START OF LIST  
1351 015616 TRAP C$BSEG ;****START OF SEGMENT****  
(3) 015616 104404
```

```
1352 015620 13:
1353 015620 012777 177776 164424 MOV #-2,@RLBA ;SET RLBA TO ALL 1'S (BIT 0-0)
1354 015626 012737 177776 002354 MOV #-2,GDDAT ;SET UP EXPECTED RESULTS
1355 015634 041337 002354 BIC (R3),GDDAT ;I'1 GDDAT
1356 015640 041377 164406 BIC (R3),@RLBA ;BIC RLBA
1357 015644 017737 164402 002356 MOV @RLBA,BDDAT ;READ RLBA
1358 015652 023737 002356 002354 CMP BDDAT,GDDAT ;BIC WORK OKAY?
1359 015660 001404 BEQ 2$ ;IF YES BRANCH
1360
1361 015662 ERRDF 10.,EM64,ERR2 ;WRONG DATA IN RLBA
(4) 015662 104455 TRAP C$ERRDF
(5) 015664 000012 .WORD 10
(5) 015666 007077 .WORD EM64
(5) 015670 010202 .WORD ERR2
1362 015672 2$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 015672 104410 TRAP C$ESCAPE
(3) 015674 000012 .WORD 10000$-.
1363
1364 015676 005723 TST (R3)+ ;GET NEXT PATTERN
1365 015700 020327 002616 CMP R3,#ENDPAT ;HAVE WE COMPLETED LIST
1366 015704 001345 BNE 1$ ;NO, GO BACK FOR NEXT
1367 015706 ENDSEG ;****END OF SEGMENT****
(3) 015706 10000$: TRAP C$ESEG
(3) 015706 104405 ;****END OF TEST****
1368 015710 ENDTST
(3) 015710 L10033: TRAP C$ETST
(3) 015710 104401
1369
1370
1371 .SBTTL **TEST 12** - BIS OF RLDA
1372
1373 015712 BGNST ;****START OF TEST****
1374
1375 015712 STARS
(2) ;*****
1376 ;TEST THAT THE 'BIS' INSTRUCTION WILL WORK ON THE DISK ADDRESS
1377 ;REGISTER. BITS 15-0 ARE TESTED WITH 4 PATTERNS, GROWING 1,
1378 ;SHIFTING 1, GROWING 0, AND SHIFTING 0.
1379 015712 STARS
(2) ;*****
1380
1381
1382 015712 012703 002410 BGNSEG MOV #BEGPAT,R3 ;GET START OF LIST
1383 015716 (3) 015716 104404 TRAP C$BSEG ;****START OF SEGMENT****
1384 015720 13:
1385 015720 005077 164330 CLR @RLDA ;CLEAR 'DA'
1386 015724 011337 002354 MOV (R3),GDDAT ;SET EXPECTED
1387 015730 051377 164320 BIS (R3),@RLDA ;BIS RLDA
1388 015734 017737 164314 002356 MOV @RLDA,BDDAT ;READ RLDA
1389 015742 023737 002356 002354 CMP BDDAT,GDDAT ;IS RLDA CORRECT
1390 015750 001404 BEQ 2$ ;IF OKAY BRANCH
1391
1392 015752 ERRDF 11.,EM65,ERR2 ;WRONG DATA IN RLDA
(4) 015752 104455 TRAP C$ERRDF
(5) 015754 000013 .WORD 11
```

```
(5) 015756 007162          .WORD EM65
(5) 015760 010202          .WORD ERR2
1393 015762          2$: ESCAPE SEG          ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 015762 104410          TRAP C$ESCAPE
(3) 015764 000012          .WORD 10000$-.
1394
1395 015766 005723          TST (R3)+          ;GET NEXT PATTERN
1396 015770 020327 002616  CMP R3,#ENDPAT    ;HAVE WE FINISHED?
1397 015774 001351          BNE 1$            ;NO GO BACK
1398 015776          ENDSEG           ;****END OF SEGMENT****
(3) 015776 10000$:
(3) 015776 104405          TRAP C$ESEG
1399 016000          ENDTST           ;****END OF TEST****
(3) 016000 L10034:
(3) 016000 104401          TRAP C$ETST
1400
1401
1402          .SBTTL **TEST 13** - BIC OF RLDA
1403
1404 016002          BGNSTST         ;****START OF TEST****
1405
1406 016002          STARS
(2)          ;:*****
1407          ;TEST THAT THE 'BIC' INSTRUCTION WORKS ON THE DISK
1408          ;ADDRESS REGISTER. ALL BITS ARE TESTED WITH FOUR
1409          ;PATTERNS: GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
1410 016002          STARS
(2)          ;:*****
1411
1412
1413 016002 012703 002410          BGNSEG MOV #BEGPAT,R3          ;GET START OF LIST
1414 016006          TRAP C$BSEG          ;****START OF SEGMENT****
(3) 016006 104404
1415 016010          1$:
1416 016010 012777 177777 164236  MOV #-1,@RLDA      ;SET RLDA TO ALL 1'S
1417 016016 012737 177777 002354  MOV #-1,GDDAT      ;SET EXPECTED DATA
1418 016024 041337 002354          BIC (R3),GDDAT     ;SET EXPECTED DATA
1419 016030 041377 164220          BIC (R3),@RLDA    ;'BIC' RLDA
1420 016034 017737 164214 002356  MOV @RLDA,BDDAT    ;READ RLDA
1421 016042 023737 002354 002356  CMP GDDAT,BDDAT    ;DID 'BIC' WORK?
1422 016050 001404          BEQ 2$            ;IF IT DID BRANCH
1423
1424 016052          ERRDF 12.,EM66,ERR2 ;WRONG DATA IN RLDA
(4) 016052 104455          TRAP C$ERDF
(5) 016054 000014          .WORD 12
(5) 016056 007243          .WORD EM66
(5) 016060 010202          .WORD ERR2
1425 016062          2$: ESCAPE SEG          ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 016062 104410          TRAP C$ESCAPE
(3) 016064 000012          .WORD 10000$-.
1426
1427 016066 005723          TST (R3)+          ;GET NEXT PATTERN
1428 016070 020327 002616  CMP R3,#ENDPAT    ;DONE?
1429 016074 001345          BNE 1$            ;NO GO BACK
1430 016076          ENDSEG           ;****END OF SEGMENT****
(3) 016076 10000$:
```

```
(3) 016076 104405 TRAP C$ESEG ;*****END OF TEST****
1431 016100 ENDTST L10035:
(3) 016100 TRAP C$SETST
(3) 016100 104401
1432
1433
1434 .SBTTL **TEST 14** - BUS RESET OF RLCS
1435 BGNIST ;*****START OF TEST****
1436 016102
1437
1438 016102 STARS
(2) ;:*****
1439 ;:TEST THAT A BUS RESET WILL CLEAR THE PROPER BITS
1440 ;:OF THE CONTROL AND STATUS REGISTER. THOSE BITS ARE
1441 ;:BITS 6-1,8,9,10,11,12,13,15. BIT 15 WILL CLEAR ONLY
1442 ;:IF BIT 14 (DRIVE ERROR IS NOT SET). BIT 0 (DRIVE READY)
1443 ;:IS A DON'T CARE. IF AT THE START UP THIS TEST BIT
1444 ;:14 (DRIVE ERROR) IS SET WE WILL INSIST IF IS THERE AFTER
1445 ;:THE 'RESET' ALONG WITH BIT 15 (COMPOSITE ERROR). BITS
1446 ;:15-10 ARE NOT WRITEABLE.
1447 016102 STARS
(2) ;:*****
1448
1449
1450 016102 SETPRI #PRI07 ;PRIORITY TO SEVEN
(3) 016102 012700 000340 MOV #PRI07,R0
(3) 016106 104441 TRAP C$SPRI
1451 016110 012777 000377 164132 MOV #377,@RLCS ;LOAD ALL RLCS LOADABLE BITS
1452 016116 012737 000200 002354 MOV #CRDY,GDDAT ;SETUP EXPECTED
1453 016124 032777 040000 164116 BIT #DERR,@RLCS ;DRIVE ERR SET?
1454 016132 001403 BEQ 1$ ;IF NOT DON'T EXPECT IT
1455 016134 052737 140000 002354 BIS #DERR!ERR,GDDAT ;IT'S SET, INIT BETTER NOT CLR
1456 016142 012700 000100 1$: MOV #100,R0 ;SET UP A WAIT LOOP
1457 016146 BRESET ;BUS RESET
(3) 016146 104433 TRAP C$RESET
1458 016150 005300 2$: DEC R0 ;WAIT IN CASE OF DRIVE ERROR
1459 016152 001376 BNE 2$
1460 016154 017737 164070 002356 MOV @RLCS,BDDAT ;READ RLCS
1461 016162 042737 000001 002356 BIC #DRDY,BDDAT ;CLEAR OUT DRDY - DON'T CARE
1462 016170 023737 002356 002354 CMP BDDAT,GDDAT ;DID INIT WORK
1463 016176 001404 BEQ 3$ ;YES, BRANCH
1464
1465 016200 ERRDF 13,EM67,ERR2 ;WRONG DATA IN RLCS
(4) 016200 104455 TRAP C$ERDF
(5) 016202 000015 .WORD 13
(5) 016204 007326 .WORD EM67
(5) 016206 010202 .WORD ERR2
1466 016210 3$:
1467 016210 ENDTST L10036: ;*****END OF TEST****
(3) 016210 TRAP C$SETST
(3) 016210 104401
1468
1469
1470 .SBTTL **TEST 15** - BUS RESET OF RLBA
1471 BGNIST ;*****START OF TEST****
1472 016212
```

```
1473
1474 016212 STARS
(2) ;*****
1475 ;TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
1476 ;BUS ADDRESS REGISTER. THE BUS ADDRESS IS LOADED WITH 177776
1477 ;AND IS EXPECTED TO BE ZERO AFTER THE RESET
1478 016212 STARS
(2) ;*****
1479
1480
1481 016212 012777 177776 164032 MOV #-2,@RLBA ;SET BA TO ALL 1'S
1482 016220 005737 002402 TST T.CNTRL ;RL11??
1483 016224 001403 BEQ 2$ ;NO
1484 016226 052777 000001 164016 BIS #1,@RLBA
1485 016234 005037 002354 2$: CLR GDDAT ;CLEAR EXPECTED DATA
1486 016240 BRESET ;ISSUE BUS INIT
(3) 016240 104433 TRAP C$RESET
1487 016242 017737 164004 002356 MOV @RLBA,BDDAT ;READ RLBA
1488 016250 001404 BEQ 1$ ;IF CLEAR BRANCH
1489
1490 016252 ERRDF 14.,EM70,ERR2 ;WRONG DATA IN RLBA
(4) 016252 104455 TRAP C$ERDF
(5) 016254 000016 .WORD 14
(5) 016256 007363 .WORD EM70
(5) 016260 010202 .WORD ERR2
1491 016262 1$:
1492
1493 016262 ENDTST ;****END OF TEST****
(3) 016262 L10037:
(3) 016262 104401 TRAP C$ETST
1494
1495
1496 .SBTTL **TEST 16** - BUS RESET OF RLDA
1497
1498 016264 BGNTST ;****START OF TEST****
1499
1500 016264 STARS
(2) ;*****
1501 ;TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
1502 ;DISK ADDRESS REGISTER. THE DISK ADDRESS IS LOADED WITH 177777
1503 ;AND IS EXPECTED TO BE ZERO AFTER THE RESET.
1504 016264 STARS
(2) ;*****
1505
1506
1507 016264 012777 177777 163762 MOV #-1,@RLDA ;SET DA TO ALL 1'S
1508 016272 005037 002354 CLR GDDAT ;CLEAR EXPECTED
1509 016276 BRESET ;ISSUE BUS INIT
(3) 016276 104433 TRAP C$RESET
1510 016300 017737 163750 002356 MOV @RLDA,BDDAT ;READ RLDA
1511 016306 001404 BEQ 1$ ;IF CLEAR BRANCH
1512
1513 016310 ERRDF 15.,EM71,ERR2 ;WRONG DATA IN RLDA
(4) 016310 104455 TRAP C$ERDF
(5) 016312 000017 .WORD 15
(5) 016314 007420 .WORD EM71
```

```
(5) 016316 010202
1514 016320
1515
1516 016320
(3) 016320
(3) 016320 104401
1517
1518
1519
1520
1521 016322
1522
1523 016322
(2)
1524
1525
1526
1527
1528
1529 016322
(2)
1530
1531
1532 016322 012737 000201 002324
1533 016330 012777 177776 163714
1534 016336 012777 177777 163710
1535 016344 013777 002324 163676
1536
1537
1538
1539 016352 022777 177776 163672
1540 016360 001412
1541
1542 016362 012737 177776 002354
1543 016370 017737 163656 002356
1544
1545 016376
(4) 016376 104455
(5) 016400 000020
(5) 016402 007455
(5) 016404 010202
1546 016406
(3) 016406 104406
1547
1548 016410 022777 177777 163636
1549 016416 001412
1550
1551 016420 012737 177777 002354
1552 016426 017737 163622 002356
1553
1554 016434
(4) 016434 104455
(5) 016436 000021
(5) 016440 007510
(5) 016442 010202
1555 016444
```

```
.WORD ERR2
1$:
ENDTST ;****END OF TEST****
L10040: TRAP C$ETST

.SBTTL **TEST 17** - UNIQUENESS OF RLCS
BGNTST ;****START OF TEST****

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

MOV #DRDY!CRDY,LDCSR ;SET DRIVE AND CONTROLLER READY
MOV #-2,@RLBA ;SET RLBA TO ALL 1'S
MOV #-1,@RLDA ;SET RLDA TO ALL 1'S
MOV LDCSR,@RLCS ;WRITE RLCS

;CHECK THAT RLBA REMAINS UNAFFECTED

CMP #-2,@RLBA ;RLBA OKAY?
BEQ 1$ ;YES, GO CHECK DA

MOV #-2,GDDAT ;SET UP EXPECTED
MOV @RLBA,BDDAT ;READ RLBA

ERRDF 16,EM72,ERR2 ;CS MODIFIED BA
TRAP C$ERDF
.WORD 16
.WORD EM72
.WORD ERR2
1$: CKLOOP ;CHECK IF /FL:LOE IS SET
TRAP C$CLP1

CMP #-1,@RLDA ;RLDA OKAY?
BEQ 2$ ;YES, CONTINUE

MOV #-1,GDDAT ;SET UP EXPECTED
MOV @RLDA,BDDAT ;READ DA

ERRDF 17,EM73,ERR2 ;CS MODIFIED DA
TRAP C$ERDF
.WORD 17
.WORD EM73
.WORD ERR2
2$:
```



```

1556
1557
1558 016444          ENDTST          ;****END OF TEST****
   (3) 016444          L10041:
   (3) 016444 104401   TRAP    C$ETST
1559
1560
1561          .SBTTL  **TEST 18** - UNIQUENESS OF RLBA
1562
1563 016446          BGNTST          ;****START OF TEST****
1564 016446          STARS
   (2)          ;:*****
1565          ;:TEST THE UNIQUENESS OF THE BUS ADDRESS REGISTER. THE
1566          ;:RLCS AND RLDA ARE LOADED WITH XXX20X AND 177777
1567          ;:RESPECTIVELY. THE RLBA IS THEN WRITTEN TO INSURE
1568          ;:THAT NEITHER THE RLCS OR RLDA ARE MODIFIED
1569          ;:BY WRITING THE RLBA.
1570 016446          STARS
   (2)          ;:*****
1571
1572
1573 016446 012737 000200 002354          MOV    #CRDY,GDDAT          ;CONTROLLER READY
1574 016454 032777 040000 163566          BIT    #DERR,@RLCS          ;IF DRIVE ERROR IS
1575 016462 001403          BEQ    99$          ;SET THEN EXPECT IT
1576 016464 052737 140000 002354          BIS    #ERR!DERR,GDDAT      ;SET WHEN WE READ IT.
1577 016472 013777 002354 163550 99$: MOV    GDDAT,@RLCS          ;LOAD RLCS
1578 016500 012777 177777 163546          MOV    #-1,@RLDA          ;LOAD RLDA
1579 016506 005077 163540          CLR    @RLBA          ;CLEAR RLBA
1580
1581          ;:CHECK IF RLCS IS OKAY
1582
1583 016512 017737 163532 002356          MOV    @RLCS,BDDAT          ;READ RLCS
1584 016520 042737 000001 002356          BIC    #DRDY,BDDAT          ;IGNORE DRIVE READY
1585 016526 023737 002356 002354          CMP    BDDAT,GDDAT          ;CS OK?
1586 016534 001404          BEQ    1$          ;YES, GO CHECK DA
1587
1588 016536          ERRDF 18.,EM74,ERR2          ;BA MODIFIED CS
   (4) 016536 104455          TRAP  C$ERDF
   (5) 016540 000022          .WORD 18
   (5) 016542 007543          .WORD EM74
   (5) 016544 010202          .WORD ERR2
1589 016546          1$: CKLOOP          ;CHECK IF /FL:LOE IS SET
   (3) 016546 104406          TRAP  C$CLP1
1590
1591 016550 022777 177777 163476          CMP    #-1,@RLDA          ;IS RLDA OKAY?
1592
1593 016556 001412          BEQ    2$          ;IF OKAY BRANCH
1594
1595 016560 012737 177777 002354          MOV    #-1,GDDAT          ;SET UP EXPECTED
1596 016566 017737 163462 002356          MOV    @RLDA,BDDAT          ;READ RLDA
1597
1598 016574          ERRDF 19.,EM75,ERR2          ;BA MODIFIED DA
   (4) 016574 104455          TRAP  C$ERDF
   (5) 016576 000023          .WORD 19
   (5) 016600 007575          .WORD EM75
   (5) 016602 010202          .WORD ERR2

```

```
1599 016604      2$:
1600 016604      ENDTST                      ;****END OF TEST****
   (3) 016604
   (3) 016604 104401      L10042:
1601                                     TRAP   C$ETST
1602
1603      .SBTTL  **TEST 19** - UNIQUENESS OF RLDA
1604
1605 016606      BGNTST                      ;****START OF TEST****
1606
1607
1608 016606      STARS
   (2)           ;:*****
1609           ;:TEST THE UNIQUENESS OF THE DISK ADDRESS REGISTER.  THE RLCS
1610           ;:AND RLBA ARE LOADED WITH XXX20X AND 177776
1611           ;:RESPECTIVELY.  THE RLDA IS THEN WRITTEN TO INSURE
1612           ;:THAT NEITHER THE RLCS OR THE RLBA ARE MODIFIED
1613           ;:BY WRITING THE RLDA.
1614 016606      STARS
   (2)           ;:*****
1615
1616
1617 016606 012737 000200 002354      MOV     #CRDY,GDDAT      ;CONTROLLER READY
1618 016614 032777 040000 163426      BIT     #DERR,@RLCS     ;IF DRIVE ERROR SET
1619 016622 001403                                     BEQ     99$              ;THEN EXPECT IT LATER
1620 016624 052737 140000 002354      BIS     #ERR!DERR,GDDAT
1621 016632 013777 002354 163410 99$:  MOV     GDDAT,@RLCS     ;LOAD CS
1622 016640 012777 177776 163404      MOV     #-2,@RLBA      ;LOAD BA WITH ALL 1'S
1623 016646 005077 163402      CLR     @RLDA          ;CLEAR RLDA
1624
1625      ;CHECK IF RLCS IS OKAY
1626
1627 016652 017737 163372 002356      MOV     @RLCS,BDDAT    ;READ RLCS
1628 016660 042737 000001 002356      BIC     #DRDY,BDDAT    ;IGNORE DRIVE READY
1629 016666 023737 002354 002356      CMP     GDDAT,BDDAT    ;RLCS OKAY?
1630 016674 001404                                     BEQ     1$              ;YES, THEN BRANCH
1631
1632 016676      ERRDF 20.,EM76,ERR2      ;DA MODIFIED CS
   (4) 016676 104455      TRAP   C$ERDF
   (5) 016700 000024      .WORD 20
   (5) 016702 007627      .WORD EM76
   (5) 016704 010202      .WORD ERR2
1633 016706      1$:  CKLOOP
   (3) 016706 104406      TRAP   C$CLP1          ;CHECK IF /FL:LOE IS SET
1634
1635 016710 022777 177776 163334      CMP     #-2,@RLBA      ;IS RLBA OKAY?
1636 016716 001412                                     BEQ     2$              ;BRANCH IF OKAY
1637
1638 016720 012737 177776 002354      MOV     #-2,GDDAT      ;SET UP EXPECTED
1639 016726 017737 163320 002356      MOV     @RLBA,BDDAT    ;READ RLBA
1640
1641 016734      ERRDF 21.,EM77,ERR2      ;DA MODIFIED BA
   (4) 016734 104455      TRAP   C$ERDF
   (5) 016736 000025      .WORD 21
   (5) 016740 007662      .WORD EM77
   (5) 016742 010202      .WORD ERR2
```

```

1642 016744          2$:
1643
1644
1645 016744          ENDTST          ;****END OF TEST****
   (3) 016744          L10043:
   (3) 016744 104401   TRAP    C$ETST
1646
1647          .SBTTL  **TEST 20** - UNIQUENESS OF RLMP
1648
1649 016746          BGNST          ;****START OF TEST****
1650
1651
1652 016746          STARS
   (2)          ;:*****
1653          ;:TEST THE UNIQUENESS OF THE MULTI-PURPOSE REGISTER
1654          ;:WE WILL WRITE THE RLCS, RLBA, AND THE RLDA, THEN THE
1655          ;:RLMP IS WRITTEN. WE THEN GO BACK AN VERIFY THE CONTENTS
1656          ;:OF THE RLCS, RLBA, RLDA.
1657 016746          STARS
   (2)          ;:*****
1658
1659
1660 016746 012737 000200 002354          MOV    #CRDY,GDDAT          ;CONTROLLER READY
1661 016754 032777 040000 163266          BIT    #DERR,@RLCS          ;IF DRIVE ERROR SET
1662 016762 001403          BEQ    99$          ;THE EXPECT IT LATER
1663 016764 052737 140000 002354          BIS    #ERR!DERR,GDDAT
1664 016772 013777 002354 163250 99$: MOV    GDDAT,@RLCS          ;LOAD CS
1665 017000 012777 177776 163244          MOV    #-2,@RLBA          ;LOAD BA WITH ALL 1'S
1666 017006 012777 177777 163240          MOV    #-1,@RLDA          ;LOAD RLDA
1667 017014 005077 163236          CLR    @RLMP          ;WRITE RLMP
1668
1669          ;CHECK IF RLCS IS OKAY
1670
1671 017020 017737 163224 002356          MOV    @RLCS,BDDAT          ;READ RLCS
1672 017026 042737 000001 002356          BIC    #DRDY,BDDAT          ;IGNORE DRIVE READY
1673 017034 023737 002354 002356          CMP    GDDAT,BDDAT          ;RLCS OKAY?
1674 017042 001404          BEQ    1$          ;YES, THEN BRANCH
1675
1676 017044          ERRDF 201.,EM44,ERR2          ;MP MODIFIED CS
   (4) 017044 104455          TRAP  C$ERDF
   (5) 017046 000311          .WORD 201
   (5) 017050 006176          .WORD EM44
   (5) 017052 010202          .WORD ERR2
1677 017054          1$: CKLOOP          ;CHECK IF /FL:LOE IS SET
   (3) 017054 104406          TRAP  C$CLP1
1678
1679 017056 022777 177776 163166          CMP    #-2,@RLBA          ;IS RLBA OKAY?
1680 017064 001412          BEQ    2$          ;BRANCH IF OKAY
1681
1682 017066 012737 177776 002354          MOV    #-2,GDDAT          ;SET UP EXPECTED
1683 017074 017737 163152 002356          MOV    @RLBA,BDDAT          ;READ RLBA
1684
1685 017102          ERRDF 211.,EM45,ERR2          ;MP MODIFIED BA
   (4) 017102 104455          TRAP  C$ERDF
   (5) 017104 000323          .WORD 211
   (5) 017106 006231          .WORD EM45
  
```

```

(5) 017110 010202
1686 017112
(3) 017112 104406
1687 017114 022777 177777 163132
1688 017122 001412
1689
1690 017124 017737 163124 002356
1691 017132 012737 177777 002354
1692
1693 017140
(4) 017140 104455
(5) 017142 000324
(5) 017144 006264
(5) 017146 010202
1694
1695 017150
1696
1697
1698 017150
(3) 017150
(3) 017150 104401
1699
1700
1701
1702 017152
1703
1704
1705
1706 017152
(2)
1707
1708
1709
1710
1711 017152
(2)
1712
1713
1714 017152 005737 002402
1715 017156 001410
1716
1717
1718 017160 004537 013414
1719 017164 000000
1720 017166 004537 014246
1721 017172
(3) 017172 104406
1722
1723 017174 004537 013114
1724
1725 017200
1726 017200
(3) 017200
(3) 017200 104401
1727
1728
  
```

```

      .WORD  ERR2
2$:  CKLOOP
      TRAP  C$CLP1      ;CHECK IF /FL:LOE IS SET
      CMP   #-1,@RLDA  ;DISK ADDRESS OKAY
      BEQ   3$          ;YES, CONTINUE

      MOV   @RLDA,BDDAT ;SET UP BAD
      MOV   #-1,GDDAT   ;SET UP EXPECTED

      ERRDF 212.,EM46,ERR2 ;MP MODIFIED DA
      TRAP  C$ERDF
      .WORD 212
      .WORD EM46
      .WORD ERR2

3$:

EI,DTST
L10044:
      TRAP  C$ETST

.SBTTL **TEST 21** - NOOP FUNCTION(RL11 ONLY)

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

STARS
;:*****
;:TEST THAT NOOP WILL FUNCTION. WE WILL ISSUE THE
;:NOOP AND WAIT FOR CONTROLLER READY TO SET. A
;:TIMEOUT OF 200 MILLISECS IS ALLOWED. DRIVE 0 IS ALWAYS
;:SELECTED SINCE THE DRIVE IS NOT NECESSARY.
STARS
;:*****

      TST   T.CNTRL      ;RLV11??
      BEQ   99$          ;YES SKIP TEST

      JSR   R5,LDFUNC    ;ISSUE FUNCTION OF FOLLOWING WORD
      NOOP0 ;NOOP(0) FUNCTION
      JSR   R5,WTCRDY    ;WAIT FOR CONTROLLER READY HIGH
2$:  CKLOOP
      TRAP  C$CLP1      ;CHECK IF /FL:LOE IS SET

      JSR   R5,CHERR     ;CHECK CONTROLLER FOR ERRORS

99$:
ENDTST
L10045:
      TRAP  C$ETST
  
```

```
1729 .SBTTL **TEST 22** - TEST NOOP DOES NOTHING
1730
1731 017202 BGNTST ;****START OF TEST****
1732
1733 017202 STARS
(2) ;*****
1734 ;TEST THAT ISSUING A NOOP FUNCTION DOES NOTHING. THIS IS DONE BY WRITING
1735 ;THE RLBA, AND RLDA, READING THE RLMP AND MAKING SURE NOTHING CHANGES.
1736 017202 STARS
(2) ;*****
1737
1738 017202 005737 002402 TST T.CNTRL ;RLV11??
1739 017206 001476 BEQ 3$
1740
1741 017210 012777 000001 163036 MOV #1,@RLDA ;LOAD DISK ADDRESS
1742 017216 012777 000002 163026 MOV #2,@RLBA ;LOAD BUS ADDRESS
1743 017224 005077 163026 CLR @RLMP
1744 017230 017737 163022 00235/ MOV @RLMP,GDDAT ;READ RLMP
1745
1746 017236 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
1747 017242 000000 NOOP0
1748 017244 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
1749 017250 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 017250 104406 TRAP C$CLP1
1750
1751 017252 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
1752 017256 ESCAPE TST ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 017256 104410 TRAP C$ESCAPE
(3) 017260 000124 .WORD L10046-.
1753
1754 017262 017737 162770 002356 MOV @RLMP,BDDAT ;READ RLMP
1755 017270 023737 002354 002356 CMP GDDAT,BDDAT ;RLMP OK?
1756 017276 001404 BEQ 1$
1757
1758 017300 ERRDF 202.,EM14,ERR2
(4) 017300 104455 TRAP C$ERDF
(5) 017302 000312 .WORD 202
(5) 017304 005276 .WORD EM14
(5) 017306 010202 .WORD ERR2
1759
1760 017310 1$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 017310 104406 TRAP C$CLP1
1761
1762 017312 012737 000002 002354 MOV #2,GDDAT ;SET UP EXP'D BA
1763 017320 017737 162726 002356 MOV @RLBA,BDDAT ;READ BA
1764 017326 023737 002354 002356 CMP GDDAT,BDDAT ;BA OK?
1765 017334 001404 BEQ 2$ ;YES
1766
1767 017336 ERRDF 203.,EM15,ERR2
(4) 017336 104455 TRAP C$ERDF
(5) 017340 000313 .WORD 203
(5) 017342 005324 .WORD EM15
(5) 017344 010202 .WORD ERR2
1768
1769 017346 2$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 017346 104406 TRAP C$CLP1
```

```

1770
1771 017350 012737 000001 002354      MOV      #1,GDDAT      ;SET UP EXP'D DA
1772 017356 017737 162672 002356      MOV      @RLDA,BDDAT  ;READ DA
1773 017364 023737 002354 002356      CMP      GDDAT,BDDAT  ;DA OKAY
1774 017372 001404
1775
1776 017374      ERRDF  204.,EM16,ERR2
   (4) 017374 104455      TRAP   C$ERDF
   (5) 017376 000314      .WORD  204
   (5) 017400 005352      .WORD  EM16
   (5) 017402 010202      .WORD  ERR2
1777
1778 017404      3$:
1779
1780 017404      ENDTST                      ;****END OF TEST****
   (3) 017404      L10046:
   (3) 017404 104401      TRAP   C$ETST
1781
1782
1783      .SBTTL **TEST 23** - TEST OF INTERRUPT
1784
1785 017406      BGNTST                      ;****START OF TEST****
1786
1787 017406      STARS
   (2)
1788      :*****
1789      :CHECK THE INTERRUPT WITH A NOOP. WE WILL SET UP THE
1790      :INTERRUPT VECTOR, LOWER THE PSW TO ZERO AND ISSUE
1791      :A NOOP. THE INTERRUPT SERVICE ROUTINE WILL SET A
1792      :FLAG UPON INTERRUPT AND RETURN IN LINE. WE WAIT 200 MILLISECONDS
1793      :LOOKING FOR THAT FLAG TO BE SET BEFORE CALLING IT
1794      :AN ERROR. IF THE INTERRUPT SENDS US TO ANOTHER
1795      :VECTOR ADDRESS THEN THE ERROR HANDLER WILL REPORT
1796      :"TRAP TO XXXX FROM YYYY" AND RETURN TO DIAG SUP MONITOR. IF THE
1797      :INTERRUPT GOES TO ABOVE 1000 WHO KNOWS WHAT WILL HAPPEN.
1798      STARS
1799      :*****
1800 017406 005737 002402      TST     T.CNTRL
1801 017412 001426      BEQ     99$
1802
1803 017414 005037 002322      CLR     INTFLG          ;CLEAR INTERRUPT OCCURRENCE FLAG
1804 017420      SETPRI #PRI00          ;SET PSW TO 0
   (3) 017420 012700 000000      MOV     #PRI00,R0
   (3) 017424 104441      TRAP   C$SPRI
1805 017426 004537 013414      JSR     R5,LDFUNC       ;ISSUE FUNCTION OF FOLLOWING WORD
1806 017432 000100      NOOP!INTEN             ;NOOP AND INTERRUPT ENABLE
1807 017434 004537 014246      JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY HIGH
1808 017440 005737 002322      TST     INTFLG         ;DID INTERRUPT OCCUR
1809 017444 001004      BNE     2$             ;IF SO BRANCH
1810 017446      ERRDF  22.,EM13,ERRO
   (4) 017446 104455      TRAP   C$ERDF
   (5) 017450 000026      .WORD  22
   (5) 017452 005244      .WORD  EM13
   (5) 017454 010152      .WORD  ERRO
1811 017456 005037 002322      2$:      CLR     INTFLG
  
```

```

1812 017462          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 017462 104406   TRAP          C$CLP1
1813 017464 004537 013114 JSR          R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
1814
1815
1816 017470          99$:
1817 017470          ENDTST
(3) 017470          L10047:          ;****END OF TEST****
(3) 017470 104401   TRAP          C$ETST
1818
1819
1820          .SBTTL  **TEST 24** - TEST PRIORITY BR LEVEL
1821
1822 017472          BGNTST          ;****START OF TEST****
1823
1824 017472          STARS
(2)          ;:*****
1825          ;TEST THAT PRIORITY GIVEN IS ACTUAL PRIORITY OF CONTROLLER. WE KNOW
1826          ;THE BOARD WILL INTERRUPT. WE WILL START TRYING TO INTERRUPT AT 7
1827          ;AND WORK DOWN TIL IT DOES INTERRUPT.
1828 017472          STARS
(2)          ;:*****
1829
1830 017472 005737 002402   TST          T.CNTRL          ;RLV11??
1831 017476 001456   BEQ          6$              ;YES, SKIP TEST
1832
1833 017500 012737 000340 002356   MOV          #340,BDDAT      ;SET UP INITIAL OF 7
1834 017506 013737 002262 002354   MOV          BPRIOR,GDDAT    ;GET GIVEN PRIORITY
1835
1836 017514          BGNSEG          ;****START OF SEGMENT****
(3) 017514 104404   TRAP          C$BSEG
1837
1838 017516 005037 002322   5$:          CLR          INTFLG          ;CLEAR INTERRUPT OCCURRENCE
1839 017522          SETPRI          BDDAT          ;SET PRIORITY
(3) 017522 013700 002356   MOV          BDDAT,RO
(3) 017526 104441   TRAP          C$SPRI
1840
1841 017530 004537 013414   JSR          R5,LDFUNC        ;ISSUE FUNCTION OF FOLLOWING WORD
1842 017534 000100   NOOPO!INTEN
1843
1844 017536 004537 014246   JSR          R5,WTCRDY        ;WAIT FOR CONTROLLER READY HIGH
1845 017542          ESCAPE          TST          ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 017542 104410   TRAP          C$ESCAPE
(3) 017544 000070   .WORD          L10050-.
1846
1847 017546 004537 013114   JSR          R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
1848 017552          ESCAPE          TST          ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 017552 104410   TRAP          C$ESCAPE
(3) 017554 000060   .WORD          L10050-.
1849
1850 017556 023737 002356 002354   CMP          BDDAT,GDDAT      ;SHOULD IT INTERRUPT
1851 017564 002012   BGE          1$              ;NO, BRANCH
1852
1853 017566 005737 002322   TST          INTFLG          ;DID INTERRUPT OCCUR
1854 017572 001004   BNE          2$              ;YES, OK
1855

```

```
1856 017574 3$: ERRDF 204.,EM17,ERR7
(4) 017574 104455 TRAP C$ERDF
(5) 017576 000314 .WORD 204
(5) 017600 005400 .WORD EM17
(5) 017602 010440 .WORD ERR7
1857
1858 017604 2$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 017604 104410 TRAP C$ESCAPE
(3) 017606 000014 .WORD 10000$-.
1859 017610 000405 BR 4$
1860 017612 005737 002322 1$: TST INTFLG ;DID INTERRUPT OCCUR
1861 017616 001772 BEQ 2$ ;NO, OK
1862 017620 000765 BR 3$ ;YES, ERROR
1863
1864 017622 ENDSEG ;****END OF SEGMENT****
(3) 017622 10000$:
(3) 017622 104405 TRAP C$ESEG
1865 017624 162737 000040 002356 4$: SUB #40,BDDAT ;NEXT LEVEL
1866 017632 100331 BPL 5$
1867
1868 017634 6$:
1869 017634 ENDTST ;****END OF TEST****
(3) 017634 L10050:
(3) 017634 104401 TRAP C$ETST
1870
1871 .SBTTL **TEST 25** - GET STATUS FUNCTION
1872
1873 017636 BGNTST ;****START OF TEST****
1874
1875
1876 017636 STARS
(2) ;*****
1877 ;TEST GET STATUS FUNCTION. THE GET STATUS FUNCTION WILL
1878 ;WORK IF DRIVE IS LOADED AND READY OR NOT. THE RLDA
1879 ;IS LOADED WITH THE GET STATUS AND MARKER BITS (BITS 1,0)
1880 ;AND THE FUNCTION IS ISSUED. WE WAIT 200 MILLISECONDS
1881 ;FOR CONTROLLER READY. VERIFY THAT NO ERRORS OCCUR.
1882 017636 STARS
(2) ;*****
1883
1884
1885 017636 012777 000013 162410 MOV #GSBIT!MK!DRST,@RLDA ;SET GET STATUS AND MARKER BIT
1886 017644 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
1887 017650 000004 GSTAT ;GET STATUS
1888 017652 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
1889 017656 2$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 017656 104406 TRAP C$CLP1
1890
1891 017660 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
1892
1893 017664 ENDTST ;****END OF TEST****
(3) 017664 L10051:
(3) 017664 104401 TRAP C$ETST
1894
1895
1896 .SBTTL **TEST 26** - GET STATUS FUNCTION INTERRUPT
```



```
1897
1898 017666          BGNTST          ;****START OF TEST****
1899
1900          ;CHECK GET STATUS UNDER INTERRUPT
1901
1902 017666 005037 002322          CLR      INTFLG          ;CLEAR INTERRUPT OCCURANCE
1903 017672          SETPRI     #PRI00          ;PSW TO LEVEL 0
(3) 017672 012700 000000          MOV      #PRI00,R0
(3) 017676 104441          TRAP     C$SPRI
1904 017700 012777 000003 162346          MOV      #GSBIT!MK,@RLDA ;SET UP DA
1905 017706 004537 013414          JSR      R5,LDFUNC        ;ISSUE FUNCTION OF FOLLOWING WORD
1906 017712 000104          GSTAT!INTEN              ;GET STATUS, INT ENABLE
1907 017714 004537 014246          JSR      R5,WTCRDY        ;WAIT FOR CONTROLLER READY HIGH
1908 017720          SETPRI     #PRI07
(3) 017720 012700 000340          MOV      #PRI07,R0
(3) 017724 104441          TRAP     C$SPRI
1909 017726 005737 002322          TST      INTFLG          ;DID INTERRUPT OCCUR
1910 017732 001004          BNE      2$              ;YES-BRANCH
1911 017734          ERRDF     28.,EM30,ERRO
(4) 017734 104455          TRAP     C$ERDF
(5) 017736 000034          .WORD   28
(5) 017740 005433          .WORD   EM30
(5) 017742 010152          .WORD   ERRO
1912 017744          2$:      CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 017744 104406          TRAP     C$CLP1
1913 017746 004537 013114          JSR      R5,CHERR         ;CHECK CONTROLLER FOR ERRORS
1914 017752 005037 002322          CLR      INTFLG          ;CLEAR INTERRUPT OCCURANCE
1915 017756          SETPRI     #PRI00          ;PSW TO LEVEL 0
(3) 017756 012700 000000          MOV      #PRI00,R0
(3) 017762 104441          TRAP     C$SPRI
1916 017764 012777 000003 162262          MOV      #GSBIT!MK,@RLDA ;SET UP DA FOR GET STATUS CMD
1917 017772 004537 013414          JSR      R5,LDFUNC        ;ISSUE FUNCTION OF FOLLOWING WORD
1918 017776 000004          GSTAT          ;GET STATUS - SHOULD NOT CAUSE AN INTERRUPT
1919 020000 004537 014246          JSR      R5,WTCRDY        ;WAIT FOR CONTROLLER READY HIGH
1920 020004          SETPRI     #PRI07
(3) 020004 012700 000340          MOV      #PRI07,R0
(3) 020010 104441          TRAP     C$SPRI
1921 020012 005737 002322          TST      INTFLG          ;DID INTERRUPT OCCUR (SHOULD NOT)
1922 020016 001404          BEQ      3$              ;NO - BRANCH (OK)
1923 020020          ERRDF     281.,EM30A,ERRO
(4) 020020 104455          TRAP     C$ERDF
(5) 020022 000431          .WORD   281
(5) 020024 005472          .WORD   EM30A
(5) 020026 010152          .WORD   ERRO
1924 020030          3$:      CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 020030 104406          TRAP     C$CLP1
1925 020032 004537 013114          JSR      R5,CHERR         ;CHECK CONTROLLER FOR ERRORS
1926 020036          ENDTST
(3) 020036          L10052:          ;****END OF TEST****
(3) 020036 104401          TRAP     C$ETST
1927
1928
1929          .SBTTL  **TEST 27** - GET STATUS FUNCTION GENERATES OPI W/O GS BIT
1930
1931 020040          BGNTST          ;****START OF TEST****
1932
```

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

```
1933 020040 STARS
(2) :*****
1934 :VERIFY THAT GET STATUS FUNCTION WILL NOT COMPLETE
1935 :WITHOUT SENDING OUT THE GET STATUS BIT IN THE RLDA.
1936 :WE SET MARKER BUT NO GET STATUS BIT IN THE RLDA AND
1937 :ISSUE A GET STATUS WE SHOULD RECIEVE AN OPI ERROR.
1938 :VERIFY THAT CONTROLLER READY SETS AND OPI SETS
1939 020040 STARS
(2) :*****
1940
1941
1942 020040 012777 000001 162206 MOV #MK,@RLDA ;SET ONLY MARKER BIT..
1943 020046 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
1944 020052 000004 GSTAT ;GET STATUS
1945 020054 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
1946 020060 032737 074000 002302 BIT #74000,E.CS
1947 020066 001405 BEQ 1$
1948 020070 012737 004045 013376 MOV #OPIERR,RESTMS
1949 020076 004537 013114 JSR R5,CHERR
1950 020102 1$: CKLOOP
(3) 020102 104406 TRAP C$CLP1
1951 020104 032737 002000 002302 BIT #OPI,E.CS ;IS OPI SET?
1952 020112 001004 BNE 2$ ;YES-BRANCH NO-CHECK TIMEOUT
1953 020114 ERRDF 29.,EM33,ERRO
(4) 020114 104455 TRAP C$ERDF
(5) 020116 000035 .WORD 29
(5) 020120 005566 .WORD EM33
(5) 020122 010152 .WORD ERRO
1954 020124 2$:
1955
1956 020124 ENDTST ;****END OF TEST****
(3) 020124 L10053:
(3) 020124 104401 TRAP C$ETST
1957
1958
1959 .SBTTL **TEST 28** - OPI UNDER INTERRUPT
1960
1961 020126 BGNTST ;****START OF TEST****
1962 020126 STARS
(2) :*****
1963 :FORCE AN OPI ERROR UNDER INTERRUPT TO VERIFY THAT
1964 :AN INTERRUPT WILL OCCUR FROM OPI. THE OPI IS FORCED
1965 :USING A GET STATUS WITHOUT THE GET STATUS BIT SET
1966 :IN RLDA.
1967 020126 STARS
(2) :*****
1968
1969
1970 020126 SETPRI #PRI00
(3) 020126 012700 000000 MOV #PRI00,R0
(3) 020132 104441 TRAP C$SPRI
1971 020134 005037 002322 CLR INTFLG
1972 020140 012777 000001 162106 MOV #MK,@RLDA ;SET ONLY MARKER BIT!!
1973 020146 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
1974 020152 000104 GSTAT!INTEN ;GET STATUS
1975 020154 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
```

```

1976 020160          SETPRI #PRI07
      (3) 020160 012700 000340      MOV #PRI07,R0
      (3) 020164 104441          TRAP C$SPRI
1977 020166 005737 002322      TST INTFLG          ;INTERRUPT OCCUR
1978 020172 001004          BNE 2$
1979 020174          ERRDF 30.,EM11,ERRO
      (4) 020174 104455          TRAP C$ERDF
      (5) 020176 000036          .WORD 30
      (5) 020200 005203          .WORD EM11
      (5) 020202 010152          .WORD ERRO
1980 020204          2$: CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 020204 104406          TRAP C$CLP1
1981 020206 032737 074000 002302      BIT #74000,E.CS
1982 020214 001405          BEQ 1$
1983 020216 012737 004045 013376      MOV #OPIERR,RESTMS
1984 020224 004537 013114          JSR R5,CHERR
1985 020230          1$: CKLOOP
      (3) 020230 104406          TRAP C$CLP1
1986 020232 032737 002000 002302      BIT #OPI,E.CS          ;IS OPI SET?
1987 020240 001004          BNE 3$          ;YES-BRANCH NO-CHECK TIMEOUT
1988 020242          ERRDF 31.,EM33,ERRO
      (4) 020242 104455          TRAP C$ERDF
      (5) 020244 000037          .WORD 31
      (5) 020246 005566          .WORD EM33
      (5) 020250 010152          .WORD ERRO
1989 020252          3$:
1990
1991 020252          ENDTST          ;****END OF TEST****
      (3) 020252          L10054:
      (3) 020252 104401          TRAP C$ETST
1992
1993          .SBTTL **TEST 29** - R. AD HEADER FUNCTION
1994
1995 020254          BGNTST          ;****START OF TEST****
1996 020254          STARS
      (2)
1997          ;:*****
1998          ;CHECK THAT READ HEADER WORKS, THAT WE CAN ISSUE
1999 020254          ;IT, GET READY BACK WITHOUT ANY ERRORS SETTING.
      (2)          ;:*****
2000
2001 020254 004537 013414          JSR R5,LDFUNC          ;ISSUE FUNCTION OF FOLLOWING WORD
2002 020260 000010          RDHDR          ;READ HEADER
2003 020262 004537 014246          JSR R5,WTCRDY          ;WAIT FOR CONTROLLER READY HIGH READY
2004 020266          2$: CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 020266 104406          TRAP C$CLP1
2005 020270 004537 013114          JSR R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
2006
2007 020274          ENDTST          ;****END OF TEST****
      (3) 020274          L10055:
      (3) 020274 104401          TRAP C$ETST
2008
2009          .SBTTL **TEST 30** - READ HEADER FUNCTION INTERRUPT
2010
2011 020276          BGNTST          ;****START OF TEST****
2012
  
```

2013 020276

(2)

2014

2015

2016 020276

(2)

2017

2018

2019 020276

(3)

(3)

2020 020304 005037 002322

2021 020310 004537 013414

2022 020314 000110

2023 020316 004537 014246

2024 020322

(3)

(3)

2025 020330 005737 002322

2026 020334 001004

2027 020336

(4)

(5)

(5)

(5)

2028 020346

(3)

2029

2030 020350 004537 013114

2031

2032 020354

(3)

(3)

2033

2034

2035

2036

2037 020356

2038

2039

2040 020356

(2)

2041

2042

2043

2044 020356

(2)

2045

2046

2047 020356 012701 000144

2048 020362 004537 013414

2049 020366 000010

2050 020370 004537 014246

2051 020374

(3)

(3)

STARS

:CHECK THAT READ HEADER WILL GENERATE AN INTERRUPT

:UPON COMPLETION WITHOUT ANY ERRORS SETTING

STARS

SETPRI #PRI00 ;PSW TO 0

MOV #PRI00,R0

TRAP C\$SPRI

CLR INTFLG ;CLEAR INTERRUPT OCCURENCE

JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD

RDHDR!INTEN ;READ HEADER, INTR. ENA

JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH

SETPRI #PRI07

MOV #PRI07,R0

TRAP C\$SPRI

TST INTFLG ;INTERRUPT HAPPEN

BNE 2\$;YES-CONTINUE

ERRDF 35,EM37,ERRO

TRAP C\$ERDF

.WORD 35

.WORD EM37

.WORD ERRO

2\$: CKLOOP ;CHECK IF /FL:LOE IS SET

TRAP C\$CLP1

JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS

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

L10056:

TRAP C\$ETST

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

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

STARS

:CHECK 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

TRAP C\$ESCAPE

.WORD L10057-

CZRLGBO RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 2-21
TEST 31 - REPEATED RD HDRS YIELD SAME CYL AND HD

SEQ 0084

```

2052
2053 020400 004537 013114      JSR   R5,CHERR      ;CHECK CONTROLLER FOR ERRORS
2054 020404          ESCAPE  TST          ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 020404 104410      TRAP   C$ESCAPE
(3) 020406 000112      .WORD  L10057-.
2055
2056 020410 013737 002310 002354  MOV   E.MP,GDDAT   ;READ FIRST HEADER (ASSUME GOOD)
2057 020416 043737 002326 002354  BIC   SECMSK,GDDAT ;MASK AWAY SECTOR BITS
2058 020424          BGNSEG          ;*****START OF SEGMENT*****
(3) 020424 104404      TRAP   C$BSEG
2059 020426          2$:
2060 020426 004537 013414      JSR   R5,LDFUNC    ;ISSUE FUNCTION OF FOLLOWING WORD
2061 020432 000010      RDHDR
2062 020434 004537 014246      JSR   R5,WTCRDY    ;WAIT FOR CONTROLLER READY HIGH
2063 020440          97$:  ESCAPE  SEG          ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 020440 104410      TRAP   C$ESCAPE
(3) 020442 000054      .WORD  10000$-.
2064
2065 020444 004537 013114      JSR   R5,CHERR      ;CHECK CONTROLLER FOR ERRORS
2066 020450          ESCAPE  SEG          ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
(3) 020450 104410      TRAP   C$ESCAPE
(3) 020452 000044      .WORD  10000$-.
2067
2068 020454 013737 002310 002356  MOV   E.MP,BDDAT   ;READ HEADER
2069 020462 043737 002326 002356  BIC   SECMSK,BDDAT ;MASK AWAY SECTOR BITS
2070 020470 023737 002354 002356  CMP   GDDAT,BDDAT  ;IS HEADER CORRECT
2071 020476 001404      BEQ   4$
2072
2073 020500          ERRDF  36.,EM41,ERR4
(4) 020500 104455      TRAP   C$ERDF
(5) 020502 000044      .WORD  36
(5) 020504 005750      .WORD  EM41
(5) 020506 010316      .WORD  ERR4
2074
2075 020510          4$:  CKLOOP
(3) 020510 104406      TRAP   C$CLP1      ;CONSTANT CYL & HS
;CHECK IF /FL:LOE IS SET
2076
2077 020512 005301          DEC   R1
2078 020514 001344          BNE   2$
;PERFORM ALL READ HDR'S
;IF NOT GO BACK AND DO ANOTHER
2079 020516          ENDSEG          ;*****END OF SEGMENT*****
(3) 020516 10000$:
(3) 020516 104405      TRAP   C$ESEG
2080 020520          ENDTST          ;*****END OF TEST*****
(3) 020520 L10057:
(3) 020520 104401      TRAP   C$ETST
2081
2082
2083 .SBTTL **TEST 32** - CHECK OF HEADER CRC
2084
2085 020522          BGNST          ;*****START OF TEST*****
2086
2087 020522          STARS
(2) ;*****
2088 ;CHECK THAT WE CAN READ THE HDCRC AFTER A
2089 ;READ HEADER AND THAT IT IS THE CORRECT CRC
2090 ;FOR THE HEADER.

```

```
2091 020522 STARS
(2) ;:*****
2092
2093
2094 020522 005037 020572 CLR 3$
2095 020526 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2096 020532 000010 RDHDR ;READ HEADER
2097 020534 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
2098 020540 ESCAPE TST ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 020540 104410 TRAP C$ESCAPE
(3) 020542 000114 .WORD L10060-.
2099
2100 020544 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2101 020550 ESCAPE TSI ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 020550 104410 TRAP C$ESCAPE
(3) 020552 000104 .WORD L10060-.
2102
2103 020554 013737 002310 020570 MOV E.MP,2$ ;READ HEADER WORD
2104 020562 004537 013772 JSR R5,SIMBCC ;GO CALCULATE CRC
2105 020566 000020 16. ;16 BITS
2106 020570 000000 2$: .WORD 0 ;HEADER GOES HERE
2107 020572 000000 3$: .WORD 0 ;START WITH 0 CRC
2108 020574 013737 002336 020620 MOV CALBCC,5$
2109 020602 013737 002312 020616 MOV E.MP1,4$ ;GET SECOND HALF
2110 020610 004537 013772 JSR R5,SIMBCC
2111 020614 000020 16.
2112 020616 000000 4$: .WORD 0
2113 020620 000000 5$: .WORD 0
2114 020622 013737 002336 002354 MOV CALBCC,GDDAT ;STORE CALCULATED CRC AS GOOD
2115 020630 013737 002314 002356 MOV E.MP2,BDDAT ;THIRD READ OF DB GETS CRC
2116 020636 023737 002354 002356 CMP GDDAT,BDDAT ;IS CRC CORRECT?
2117 020644 001404 BEQ 6$ ;IF SO CONTINUE
2118
2119 020646 ERRDF 37.,EM42,ERR4
(4) 020646 104455 TRAP C$ERDF
(5) 020650 000045 .WORD 37
(5) 020652 006041 .WORD EM42
(5) 020654 010316 .WORD ERR4
2120 6$:
2121
2122 020656 ENDTST ;****END OF TEST****
(3) 020656 L10060:
(3) 020656 104401 TRAP C$ETST
2123
2124
2125 .SBTTL **TEST 33** - CHECK CONSECUTIVE HEADERS
2126
2127 020660 BGNTST ;****START OF TEST****
2128
2129
2130 020660 STARS
(2) ;:*****
2131 ;CHECK THAT THE HEADERS ARE CONSECUTIVE. WE WILL DO
2132 ;40 (FORTY) READ HEADERS AND STORE EACH. AFTER WE HAVE
2133 ;READ THE FORTIETH HEADER WE WILL VERIFY THAT
2134 ;THEY CAME IN SEQUENTIAL, THAT 0 FOLLOWS 39.
```

```
2135 ;THAT THERE WERE NO ERRORS.
2136 020660 STARS
(2) ;:*****
2137
2138
2139 020660 005037 002360 CLR FIRST ;CLEAR FIRST READ DONE FLAG
2140 020664 012703 003266 MOV #HDRBUF,R3 ;STORE HEADERS
2141 020670 012701 000050 MOV #40.,R1 ;FORTY HEADERS
2142 020674 012737 000210 002270 MOV #RDHDR!CRDY,B.CS
2143 020702 053737 002266 002270 BIS DRIVE,B.CS
2144 020710 013777 002270 161332 MOV B.CS,@RLCS
2145 020716 042777 000200 161324 2$: BIC #200,@RLCS
2146 020724 032777 000200 161316 1$: BIT #200,@RLCS ;DONE?
2147 020732 001774 BEQ 1$
2148 020734 017723 161310 MOV @RLCS,(R3)+
2149 020740 017723 161312 MOV @RLMP,(R3)+
2150 020744 017723 161306 MOV @RLMP,(R3)+
2151 020750 017723 161302 MOV @RLMP,(R3)+
2152 020754 005301 DEC R1 ;HAVE WE READ FORTY HEADERS
2153 020756 001357 BNE 2$ ;GO BACK UNTIL FOURTY DONE
2154 020760 012703 003266 MOV #HDRBUF,R3 ;GET LIST OF HEADERS
2155 020764 012701 000050 MOV #40.,R1 ;CHECK FORTY OF THEM
2156 020770 011337 002302 MOV (R3),E.CS
2157 020774 005737 002302 TST E.CS
2158 021000 100016 BPL 99$
2159 021002 012737 004304 013376 MOV #RHDMS,RESTMS
2160 021010 005723 TST (R3)+
2161 021012 012337 002310 MOV (R3)+,E.MP
2162 021016 012337 002312 MOV (R3)+,E.MP1
2163 021022 012337 002314 MOV (R3)+,E.MP2
2164 021026 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2165 021032 000137 021174 JMP 7$
2166 021036 005723 99$: TST (R3)+
2167 021040 011337 002356 MOV (R3),BDDAT ;GET HEADER
2168 021044 005737 002360 TST FIRST ;IS THIS FIRST READ?
2169 021050 001007 BNE 4$ ;NO, BRANCH
2170 021052 012737 000001 002360 MOV #1,FIRST ;SET FIRST READ DONE FLAG
2171 021060 013737 002356 002354 3$: MOV BDDAT,GDDAT ;SET UP NEXT READ EXPECTED
2172 021066 000435 BR 6$ ;GO SEE IF TEST IS DONE
2173 021070 005237 002354 4$: INC GDDAT ;INCREMENT EXP'D HEADER
2174 021074 023737 002356 002354 CMP BDDAT,GDDAT ;IS NEW HEADER SEQUENTIAL?
2175 021102 001766 BEQ 3$ ;YES THEN BRANCH
2176 021104 033737 002326 002356 BIT SECMSK,BDDAT ;IS NEW HEADER ZERO?
2177 021112 001015 BNE 5$ ;NO, THEN ERROR GO REPORT IT
2178 021114 013737 002354 002340 MOV GDDAT,TEMP2 ;YES, CHECK IF LAST HEADER WAS
2179 021122 043737 002362 002340 BIC CYLSK,TEMP2 ;MAX ADDRESS, IF SO BRANCH
2180 021130 023737 002364 002340 CMP MXSEC1,TEMP2 ;STORE NEW DATA AS OLD
2181 021136 001750 BEQ 3$ ;AND PERFORM NEW RD HDR
2182 021140 043737 002326 002354 BIC SECMSK,GDDAT ;EXPECTING ZERO SECTOR
2183
2184 021146 5$:
2185
2186 021146 005037 002360 CLR FIRST ;ERROR WILL MAKE US MISS
2187 ;NEXT SECTOR SEQUENTIALLY
2188 ;START OVER; CLEAR FIRST FLAG
2189 021152 ERRDF 38.,EM43,ERR2
```

```
(4) 021152 104455          TRAP    C$ERDF
(5) 021154 000046          .WORD  38
(5) 021156 006077          .WORD  EM43
(5) 021160 010202          .WORD  ERR2
2190 021162                6$:    CKLOOP
(3) 021162 104406          TRAP    C$CLP1          ;CHECK IF /FL:LOE IS SET
2191
2192 021164 062703 000006  ADD     #6,R3
2193 021170 005301          DEC     R1              ;HAVE WE DONE THIS ENOUGH
2194 021172 001321          BNE    99$              ;NO, GO BACK DO IT AGAIN
2195 021174                7$:
2196 021174          ENDTST                ;****END OF TEST****
(3) 021174          L10061:
(3) 021174 104401          TRAP    C$ETST
2197
2198
2199          .SBTTL  **TEST 34** - SEEK FUNCTION
2200
2201 021176          BGNTST                ;****START OF TEST****
2202 021176          STARS
(2)          ;:*****
2203          ;CHECK THE SEEK FUNCTION RETURNS CONTROLLER READY
2204          ;WITH NO ERRORS. WE ISSUE A ONE TRACK IN WORD SEEK.
2205          ;WE DO NOT CHECK THE RESULT FOR POSITION
2206 021176          STARS
(2)          ;:*****
2207
2208
2209 021176 012777 000205 161050  MOV     #BIT7!MK!SIGN,@RLDA ;SET UP DA-DIFF=1,MARKER,TOWARDS
2210 021204 004537 013414          JSR    R5,LDFUNC        ;ISSUE FUNCTION OF FOLLOWING WORD
2211 021210 000006          SEEK
2212 021212 004537 014246          JSR    R5,WTCRDY        ;WAIT FOR CONTROLLER READY HIGH
2213 021216 012737 000010 002406  MOV     #8,DLYCNT        ;INITIALIZE DELAY COUNT
2214 021224          WAIT1: DELAY    #250            ;IMPLEMENT TIME DELAY
(2) 021224 012727 000372          MOV     #250.,(PC)+
(2) 021230 000000          .WORD  0
(2) 021232 013727 002116          MOV     L$DLY,(PC)+
(2) 021236 000000          .WORD  0
(2) 021240 005367 177772          DEC     -6(PC)
(2) 021244 001375          BNE    -4
(2) 021246 005367 177756          DEC     -22(PC)
(2) 021252 001367          BNE    -20
2215 021254 005337 002406          DEC     DLYCNT          ;DECREMENT DELAY COUNT
2216 021260 001361          BNE    WAIT1           ;BRANCH IF DELAY NOT EXPIRED
2217 021262                2$:    CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 021262 104406          TRAP    C$CLP1
2218 021264 004537 013114          JSR    R5,CHERR        ;CHECK CONTROLLER FOR ERRORS
2219
2220          ENDTST                ;****END OF TEST****
(3) 021270          L10062:
(3) 021270 104401          TRAP    C$ETST
2221
2222
2223          .SBTTL  **TEST 35** - CHECK DRIVE READY ON SEEK
2224
2225 021272          BGNTST                ;****START OF TEST****
```



```
2226
2227
2228 021272          STARS
(2)                ;:*****
2229                ;CHECK THE SEEK FUNCTION RETURNS DRIVE READY WITH
2230                ;NO ERRORS. WE ISSUE A ONE TRACK INWARD SEEK. WE DO
2231                ;NOT CHECK THE RESULT FOR POSITION
2232 021272          STARS
(2)                ;:*****
2233
2234
2235
2236 021272 012777 000201 160754      MOV    #BIT7.MK,@RLDA    ;SET DA, MARKER, DIFF=1.
2237 021300 004537 013414              JSR    R5,LDFUNC        ;ISSUE FUNCTION OF FOLLOWING WORD
2238 021304 000006                      SEEK                    ;SEEK
2239 021306 004537 014246              JSR    R5,WTCRDY        ;WAIT FOR CONTROLLER READY HIGH
2240 021312                      CKLOOP                      ;CHECK IF /FL:LOE IS SET
(3) 021312 104406                      TRAP   C$CLP1
2241
2242 021314 004537 013114              JSR    R5,CHERR         ;CHECK CONTROLLER FOR ERRORS
2243 021320                      CKLOOP                      ;CHECK IF /FL:LOE IS SET
(3) 021320 104406                      TRAP   C$CLP1
2244
2245 021322 004537 014160              JSR    R5,WTD RDY       ;WAIT FOR DRIVE READY
2246 021326                      CKLOOP                      ;CHECK IF /FL:LOE IS SET
(3) 021326 104406                      TRAP   C$CLP1
2247
2248 021330 004537 013114              JSR    R5,CHERR         ;CHECK CONTROLLER FOR ERRORS
2249
2250 021334          ENDTST
(3) 021334          L10063:
(3) 021334 104401          TRAP   C$ETST
2251
2252
2253          .SBTTL  **TEST 36** - SEEK FUNCTION INTERRUPT
2254
2255 021336          BGNTST
2256
2257
2258 021336          STARS
(2)                ;:*****
2259                ;CHECK THAT CONTROLLER READY RESETTING WHEN THE SEEK IS
2260                ;INITIATED CAUSES AN INTERRUPT BUT DRIVE READY WILL
2261                ;NOT. WE ALSO MONITOR FOR ANY ERROR BITS SETTING.
2262 021336          STARS
(2)                ;:*****
2263
2264
2265
2266
2267 021336 005037 002322              CLR    INTFLG
2268 021342          SETPRI #PRI00                ;SET PSW TO 0
(3) 021342 012700 000000              MOV    #PRI00,R0
(3) 021346 104441          TRAP   C$SPRI
2269 021350 012777 000205 160676      MOV    #BIT7!MK!SIGN,@RLDA ;SET UP RLDA
2270 021356 004537 013414              JSR    R5,LDFUNC        ;ISSUE FUNCTION OF FOLLOWING WORD
```

```
2271 021362 000106          SEEK!INTEN          ;SEEK AND INTR. ENA.
2272 021364 004537 014246    JSR      R5,WTCRDY  ;WAIT FOR CONTROLLER READY HIGH
2273 021370 000240          NOP                ;
2274 021372 005737 002322    1$:  TST      INTFLG  ;DID INTERRUPT OCCUR
2275 021376 001004          BNE      2$        ;YES, GO CHECK DRDY
2276 021400          ERRDF  40.,EM47,ERRO
      (4) 021400 104455    TRAP    C$ERDF
      (5) 021402 000050    .WORD  40
      (5) 021404 006317    .WORD  EM47
      (5) 021406 010152    .WORD  ERRO
2277 021410          2$:  CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 021410 104406    TRAP    C$CLP1
2278
2279
2280 021412 004537 013114    JSR      R5,CHERR   ;CHECK CONTROLLER FOR ERRORS
2281 021416          CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 021416 104406    TRAP    C$CLP1
2282
2283 021420 005037 002322    CLR      INTFLG     ;CLEAR INTERRUPT OCCURANCE
2284
2285
2286 021424 004537 014160    5$:  JSR      R5,WTD RDY ;WAIT FOR DRIVE READY
2287 021430          CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 021430 104406    TRAP    C$CLP1
2288
2289 021432          SETPRI  #PRI07
      (3) 021432 012700 000340  MOV     #PRI07,R0
      (3) 021436 104441    TRAP    C$SPRI
2290 021440 005737 002322    TST     INTFLG     ;DID DRIVE READY CAUSE INTERRUPT
2291 021444 001404          BEQ     6$        ;NO, CONTINUE
2292
2293 021446          ERRDF  42.,EM52,ERRO
      (4) 021446 104455    TRAP    C$ERDF
      (5) 021450 000052    .WORD  42
      (5) 021452 006350    .WORD  EM52
      (5) 021454 010152    .WORD  ERRO
2294 021456          6$:  CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 021456 104406    TRAP    C$CLP1
2295
2296 021460          ENDTST          ;****END OF TEST****
      (3) 021460          L10064:
      (3) 021460 104401    TRAP    C$ETST
2297
2298
2299          .SBTTL  **TEST 37** - TEST DIFFERENCE WORD TRANSMISSION
2300
2301 021462          BGNTST          ;****START OF TEST****
2302
2303
2304
2305
2306 021462          STARS
      (2)
2307          ;:*****
2308          ;:VERIFY THAT THE DIFFERENCE WORD LOADS AND IS
2309          ;:TRANSMITTED CORRECTLY. WE WILL ISSUE SEEKS WITH THE
          ;:DIFFERENCE WORD CONTAINING ALL OF THE BIT PATTERNS FLOATING 1.
```

```
2310 ;GROWING 1, GROWING 0 AND SHITING 0. THE SEEK WILL
2311 ;START FROM TRACK 0 EACH TIME AND WILL RETURN THERE
2312 ;EACH, THUS BOTH DIRECTIONS FOR PATTERNS WILL BE CHECKED.
2313 ;DEAD HEADERS ARE USED TO VERIFY THE SEEK CORRECTNESS.
2314 ;ERRORS ARE MONITORED AND REPORTED.
2315 021462 STARS
(2) ;*****
2316
2317
2318 021462 012703 002620 BGNSSEG MOV #SKLST,R3 ;GET LIST OF DIFFERENCE WORDS
2319 021466 104404 TRAP CSBSEG ;****START OF SEGMENT****
(3)
2320 021470 1$: JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2321 021470 004537 013414 RDHDR ;READ HEADER
2322 021474 000010 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
2323 021476 004537 014246 98$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) TRAP CSCLP1
2325
2326 021504 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2327 021510 104406 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) TRAP CSCLP1
2328
2329 021512 013737 002310 002356 MOV E,MP,BDDAT ;READ HEADER
2330 021520 043737 002326 002356 BIC SECMSK,BDDAT ;CLEAR OUT SECTOR
2331 021526 001462 BEQ 99$ ;IF ON TRACK ZERO, H.S. ZERO, OK
2332
2333 ;NOT ON TRACK ZERO CALCULATE DIFFERENCE WORD AND PUT IT BACK
2334 ;ON ZERO.
2335
2336 021530 042737 000100 002356 BIC #RHHS,BDDAT ;CLEAR OUT HEAD SELECT
2337 021536 013777 002356 160510 MOV BDDAT,@RLDA ;PUT CYLINDER AS DIFFERENCE WORD
2338 021544 052777 000001 160502 BIS #MK,@RLDA ;SET MARKER BIT
2339 021552 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2340 021556 000006 SEEK ;SEEK
2341 021560 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
2342 021564 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) TRAP CSCLP1
2343
2344 021566 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2345 021572 104406 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) TRAP CSCLP1
2346
2347 021574 004537 014160 89$: JSR R5,WTCRDY ;WAIT FOR DRIVE READY
2348 021600 104406 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) TRAP CSCLP1
2349
2350 021602 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2351 021606 104406 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) TRAP CSCLP1
2352
2353 021610 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2354 021614 000010 RDHDR ;READ HEADER
2355 021616 004537 014246 96$: JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
2356 021622 104406 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) TRAP CSCLP1
```

CZRLGB0 RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 2-28
TEST 37 - TEST DIFFERENCE WORD TRANSMISSION

SEQ 0091

2357										
2358	021624	004537	013114							
2359	021630									
(3)	021630	104406								
2360										
2361	021632	005037	002354							
2362	021636	013737	002356	002370						
2363	021644	013737	002310	002356						
2364	021652	043737	002326	002356						
2365	021660	001404								
2366										
2367	021662									
(4)	021662	104455								
(5)	021664	000053								
(5)	021666	006420								
(5)	021670	010244								
2368	021672									
(3)	021672	104406			5\$:					
2369										
2370	021674	011377	160354							
2371	021700	052777	000005	160346	99\$:					
2372	021706	004537	013414							
2373	021712	000006								
2374	021714	004537	014246							
2375	021720									
(3)	021720	104406								
2376										
2377	021722	004537	013114							
2378	021726									
(3)	021726	104406								
2379										
2380	021730	004537	014160							
2381	021734									
(3)	021734	104406			87\$:					
2382										
2383	021736	004537	013114							
2384	021742									
(3)	021742	104406								
2385										
2386	021744	004537	013414							
2387	021750	000010								
2388										
2389	021752	004537	014246							
2390	021756									
(3)	021756	104406								
2391										
2392	021760	004537	013114							
2393	021764									
(3)	021764	104410								
(3)	021766	000106								
2394										
2395	021770	011337	002354							
2396	021774	011337	002370							
2397	022000	013737	002310	002356	8\$:					
2398	022006	043737	002326	002356						
2399	022014	023737	002354	002356						

```

2400 022022 001404          BEQ      9$          ;TRACK, IF SO, GO GET NEXT
2401
2402 022024          ERRDF   44.,EM54,ERR3
(4) 022024 104455          TRAP   C$ERDF
(5) 022026 000054          .WORD  44
(5) 022030 006420          .WORD  EM54
(5) 022032 010244          .WORD  ERR3
2403 022034          9$:      CKLOOP
(3) 022034 104406          TRAP   C$CLP1          ;CHECK IF /FL:LOE IS SET
2404
2405 022036 005723          TST    (R3)+          ;BUMP PATTERN
2406 022040 023727 002400 000001  CMP    T.DRIVE,#1
2407 022046 001005          BNE    2$
2408 022050 020327 002720  CMP    R3,#SKEND
2409 022054 001407          BEQ    10$
2410 022056 000137 021470  JMP    1$
2411
2412 022062 020327 002762  2$:    CMP    R3,#SKEEND
2413 022066 001402          BEQ    10$
2414 022070 000137 021470  JMP    1$
2415
2416 022074          10$:
2417
2418 022074          ENDSEG
(3) 022074 10000$:          ;****END OF SEGMENT****
(3) 022074 104405          TRAP   C$ESEG
2419 022076          ENDTST
(3) 022076 100065$:          ;****END OF TEST****
(3) 022076 104401          TRAP   C$ETST
2420
2421
2422          .SBTTL  **TEST 38** - VERIFY HEAD SELECT 0 VIA RD HDR
2423
2424 022100          BGNST          ;****START OF TEST****
2425
2426          ;
2427
2428 022100          STARS
(2)          ;:*****
2429          ;CHECK THAT WE CAN SELECT HEAD SELECT ZERO.  ISSUE
2430          ;SEEK TO HEAD SELECT 0 AND VERIFY WITH READ HEADER.
2431 022100          STARS
(2)          ;:*****
2432
2433 022100 012777 000001 160146 99$:    MOV    #MK,@RLDA          ;SET MARKER IN RLDA
2434 022106 005037 002354          CLR    GDDAT          ;SET EXPECTED
2435          ;LOAD HS=0 INTO RLDA
2436          2$:
2437 022112 004537 013414          JSR    R5,LDFUNC          ;ISSUE FUNCTION OF FOLLOWING WORD
2438 022116 000006          SEEK          ;SEEK
2439 022120 004537 014246          JSR    R5,WTCRDY          ;WAIT FOR CONTROLLER READY HIGH
2440 022124          CKLOOP
(3) 022124 104406          TRAP   C$CLP1          ;CHECK IF /FL:LOE IS SET
2441
2442 022126 004537 013114          JSR    R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
2443 022132          CKLOOP          ;CHECK IF /FL:LOE IS SET
  
```

```

(3) 022132 104406 TRAP C$CLP1
2444
2445 022134 004537 014160 JSR R5,WTD RDY ;WAIT FOR DRIVE READY
2446 022140 89$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 022140 104406 TRAP C$CLP1
2447
2448 022142 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2449 022146 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 022146 104406 TRAP C$CLP1
2450
2451 022150 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2452 022154 000010 RDHDR ;READ HEADER
2453 022156 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
2454 022162 96$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 022162 104406 TRAP C$CLP1
2455
2456 022164 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2457 022170 ESCAPE TST ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 022170 104410 TRAP C$ESCAPE
(3) 022172 000036 .WORD L10066-.
2458
2459 022174 013737 002310 002356 MOV E,MP,BDDAT ;READ HEADER FOR HEAD SELECT
2460 022202 042737 177677 002356 BIC #177677,BDDAT ;MASK ONLY HEAD SELECT
2461 022210 023737 002354 002356 CMP GDDAT,BDDAT ;COMPARE HEAD SELECTS
2462 022216 001404 BEQ $$ ;IF EQUAL CONTINUE
2463
2464 022220 ERRDF 45.,EM55,ERR4
(4) 022220 104455 TRAP C$ERDF
(5) 022222 000055 .WORD 45
(5) 022224 006457 .WORD EM55
(5) 022226 010316 .WORD ERR4
2465 022230 5$:
2466
2467 022230 ENDTST ;****END OF TEST****
(3) 022230 L10066:
(3) 022230 104401 TRAP C$ETST
2468
2469
2470 .SBTTL **TEST 39** - VERIFY HEAD SELECT 1 VIA RD HDR
2471
2472 022232 BGNTST ;****START OF TEST****
2473
2474
2475 022232 STARS
(2) ;:*****
2476 ;:CHECK THAT WE CAN SELECT HEAD SELECT ONE. ISSUE
2477 ;:SEEK TO HEAD SELECT 1 AND VERIFY WITH READ HEADER.
2478 022232 STARS
(2) ;:*****
2479
2480
2481 022232 012777 000001 160014 99$: MOV #MK,@RLDA ;SET MARKER IN RLDA
2482 022240 052777 000020 160006 BIS #DAHS,@RLDA ;LOAD HS=1 INTO RLDA
2483 022246 004537 013414 2$: JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2484 022252 000006 SEEK ;SEEK
2485 022254 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
  
```

```
2486 022260          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 022260 104406   TRAP          C$CLP1
2487
2488 022262 004537 013114 JSR          R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
2489 022266          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 022266 104406   TRAP          C$CLP1
2490
2491 022270 004537 014160 JSR          R5,WTDRDY         ;WAIT FOR DRIVE CLEAR
2492 022274          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 022274 104406   TRAP          C$CLP1
2493
2494 022276 004537 013114 JSR          R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
2495 022302          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 022302 104406   TRAP          C$CLP1
2496
2497 022304 004537 013414 JSR          R5,LDFUNC         ;ISSUE FUNCTION OF FOLLOWING WORD
2498 022310 000010   RDHDR          ;READ HEADER
2499 022312 004537 014246 JSR          R5,WTDRDY         ;WAIT FOR CONTROLLER READY HIGH
2500 022316          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 022316 104406   TRAP          C$CLP1
2501
2502 022320 004537 013114 JSR          R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
2503 022324          ESCAPE          ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 022324 104410   TRAP          C$ESCAPE
(3) 022326 000044   .WORD        L10067-.
2504
2505 022330 013737 002310 002356 MOV          E,MP,BDDAT        ;READ HEADER
2506 022336 042737 177677 002356 BIC          #177677,BDDAT     ;MASK FOR H.S.
2507 022344 012737 000100 002354 MOV          #RHHS,GDDAT      ;SET EXPECTED
2508 022352 023737 002354 002354 CMP          GDDAT,BDDAT      ;CORRECT HEAD
2509 022360 001404   BEQ          $$              ;YES, CONTINUE
2510
2511 022362          ERRDF          46.,EM55,ERR4
(4) 022362 104455   TRAP          C$ERDF
(5) 022364 000056   .WORD        46
(5) 022366 006457   .WORD        EM55
(5) 022370 010316   .WORD        ERR4
2512 022372          S$:
2513
2514 022372          ENDTST          ;****END OF TEST****
(3) 022372          L10067:
(3) 022372 104401   TRAP          C$ETST
2515
2516
2517          .SBTTL  **TEST 40** - VERIFY HEAD SELECT 0 VIA GET STATUS
2518
2519 022374          BGNTST          ;****START OF TEST****
2520
2521 022374          STARS
(2)          ;:*****
2522          ;CHECK THAT WE CAN READ BACK HEAD SELECT 0 WITH
2523          ;A GET STATUS FUNCTION. SELECT H.S. 0 WITH A SEEK
2524          ;VERIFY WITH GET STATUS
2525 022374          STARS
(2)          ;:*****
2526
```

```

2527 022374 012777 000001 157652      MOV      #MK,@RLDA      ;SET MARKER IN RLDA
2528                                     ;LOAD HS=0 INTO RLDA
2529 022402 005037 002354      2$: CLR      GDDAT      ;SET UP EXP'D
2530 022406 004537 013414      3$: JSR      R5,LDFUNC  ;ISSUE FUNCTION OF FOLLOWING WORD
2531 022412 000006                                     ;SEEK
2532 022414 004537 014246      JSR      R5,WTCRDY    ;WAIT FOR CONTROLLER READY HIGH
2533 022420                                     ;CHECK IF /FL:LOE IS SET
(3) 022420 104406      CKLOOP
TRAP      C$CLP1
2534
2535 022422 004537 013114      JSR      R5,CHERR     ;CHECK CONTROLLER FOR ERRORS
2536 022426                                     ;CHECK IF /FL:LOE IS SET
(3) 022426 104406      CKLOOP
TRAP      C$CLP1
2537
2538 022430 004537 014160      JSR      R5,WTCRDY    ;WAIT FOR DRIVE READY
2539 022434                                     ;CHECK IF /FL:LOE IS SET
(3) 022434 104406      CKLOOP
TRAP      C$CLP1
2540
2541 022436 004537 013114      JSR      R5,CHERR     ;CHECK CONTROLLER FOR ERRORS
2542 022442                                     ;CHECK IF /FL:LOE IS SET
(3) 022442 104406      CKLOOP
TRAP      C$CLP1
2543
2544 022444 012777 000003 157602      MOV      #GSBIT!MK,@RLDA ;SET UP FOR GET STATUS IN DA
2545 022452 004537 013414      JSR      R5,LDFUNC  ;ISSUE FUNCTION OF FOLLOWING WORD
2546 022456 000004                                     ;GET STATUS
2547 022460 004537 014246      JSR      R5,WTCRDY    ;WAIT FOR CONTROLLER READY HIGH
2548 022464                                     ;CHECK IF /FL:LOE IS SET
(3) 022464 104406      CKLOOP
TRAP      C$CLP1
2549
2550 022466 004537 013114      JSR      R5,CHERR     ;CHECK CONTROLLER FOR ERRORS
2551 022472                                     ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 022472 104410      ESCAPE  TST
(3) 022474 000036      TRAP    C$ESCAPE
        .WORD  L10070-.
2552
2553 022476 013737 002310 002356      MOV      E.MP,BDDAT   ;READ STATUS FOR HEAD SELECT BIT
2554 022504 042737 177677 002356      BIC      #177677,BDDAT ;LEAVE ONLY H.S. BIT
2555 022512 023737 002354 002356      CMP      GDDAT,BDDAT  ;IS HEAD SELECT CORRECT?
2556 022520 001404                                     BEQ      6$           ;YES, CONTINUE
2557
2558 022522                                     ERRDF   47.,EM56,ERR4
(4) 022522 104455      TRAP    C$ERDF
(5) 022524 000057      .WORD  47
(5) 022526 006512      .WORD  EM56
(5) 022530 010316      .WORD  ERR4
2559 022532      6$:
2560
2561 022532      ENDTST ;****END OF TEST****
(3) 022532      L10070:
(3) 022532 104401      TRAP    C$ETST
2562
2563
2564      .SBTTL **TEST 41** - VERIFY HEAD SELECT 1 VIA GET STATUS
2565
2566 022534      BGNTST ;****START OF TEST****
2567
2568 022534      STARS
(2)      ;:*****
  
```



```
2569 ;CHECK THAT WE CAN READ BACK HEAD SELECT 1 WITH A GET
2570 ;STATUS FUNCTION. SELECT H.S. 1 WITH A SEEK AND VERIFY WITH
2571 ;GET STATUS
2572 022534 STARS
(2) ;:*****
2573
2574
2575 022534 012777 000001 157512 MOV #MK,@RLDA ;SET MARKER IN RLDA
2576 022542 052777 000020 157504 BIS #DAHS,@RLDA ;LOAD HS=1 INTO RLDA
2577 022550 012737 000100 002354 2$: MOV #STHS,GDDAT ;SET UP EXP'D
2578 022556 004537 013414 3$: JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2579 022562 000006 SEEK ;SEEK
2580 022564 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
2581 022570 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 022570 104406 TRAP C$CLP1
2582
2583 022572 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2584 022576 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 022576 104406 TRAP C$CLP1
2585
2586 022600 004537 014160 JSR R5,WTCRDY ;WAIT FOR DRIVE READY
2587 022604 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 022604 104406 TRAP C$CLP1
2588
2589 022606 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2590 022612 CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 022612 104406 TRAP C$CLP1
2591
2592 022614 012777 000003 157432 MOV #GSBIT!MK,@RLDA ;SET UP FOR GET STATUS IN DA
2593 022622 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2594 022626 000004 GSTAT ;GET STATUS
2595 022630 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
2596 022634 ESCAPE TST ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 022634 104410 TRAP C$ESCAPE
(3) 022636 000046 .WORD L10071-.
2597
2598 022640 004537 013114 JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
2599 022644 ESCAPE TST ;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3) 022644 104410 TRAP C$ESCAPE
(3) 022646 000036 .WORD L10071-.
2600
2601 022650 013737 002310 002356 MOV E.MP,BDDAT ;READ STATUS FOR HEAD SELECT BIT
2602 022656 042737 177677 002356 BIC #177677,BDDAT ;LEAVE ONLY H.S. BIT
2603 022664 023737 002354 002356 CMP GDDAT,BDDAT ;IS HEAD SELECT CORRECT?
2604 022672 001404 BEQ 6$ ;YES, CONTINUE
2605
2606 022674 ERRDF 48.,EM56,ERR4
(4) 022674 104455 TRAP C$ERRDF
(5) 022676 000060 .WORD 48
(5) 022700 006512 .WORD EM56
(5) 022702 010316 .WORD ERR4
2607 022704 6$:
2608
2609 022704 ENDTST ;****END OF TEST****
(3) 022704 L10071:
(3) 022704 104401 TRAP C$ETST
```

```
2610
2611
2612 .SBTTL **TEST 42** - TEST TIME AT WHICH DIF WD GETS TRANSMITTED
2613
2614 022706 BGNTST ;****START OF TEST****
2615
2616
2617 022706 STARS
(2) ;:*****
2618 ;:VERIFY THAT THE DIFFERENCE WORD ON A SEEK IS
2619 ;:TRANSMITTED PRIOR TO CONTROLLER READY SETTING. THIS
2620 ;:IS DONE BY SETTING A KNOWN DIFFERENCE WORD IN
2621 ;:THE RLDA ISSUING A A SEEK, WAITING FOR CONTROLLER READY
2622 ;:(BUT NOT DRIVE READY), WRITING A DIFFERENT RLDA AND WAITING
2623 ;:FOR DRIVE READY. THE RESULTANT POSITION SHOULD BE THAT
2624 ;:OF THE FIRST RLDA ONLY.
2625 022706 STARS
(2) ;:*****
2626
2627
2628 022706 004537 013414 JSR R5,LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
2629 022712 000010 RDHDR ;READ HEADER
2630 022714 004537 014246 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
2631 022720 99$: CKLOOP ;CHECK IF /FL:LOE IS SET
(3) 022720 104406 TRAP C$CLP1
2632
2633 022722 004537 013114 JSR R5,CHEP0 ;CHECK CONTROLLER FOR ERRORS
2634 022726 (3) 022726 104406 TRAP C$CLP1 ;CHECK IF /FL:LOE IS SET
2635
2636 022730 013737 002310 002354 MOV E.MP,GDDAT ;READ HEADER
2637 022736 043737 002326 002354 BIC SECMSK,GDDAT ;CLEAR SECTOR BITS
2638 022744 012777 000001 157302 MOV #MK,@RLDA ;SET MARKER IN RLDA
2639 022752 032737 000100 002354 BIT #RHHS,GDDAT ;TEST H.S.
2640 022760 001403 BEQ 2$ ;IF ZERO, CONTINUE
2641 022762 052777 000020 157264 BIS #DAHS,@RLDA ;ONE, SET SO WE WILL REMAIN THERE
2642 022770 013737 002354 002346 2$: MOV GDDAT,TMPO ;STORE HEADER
2643 022776 042737 000100 002346 BIC #RHHS,TMPO ;CLEAR H.S. FROM STORED WORD
2644 023004 023727 002400 00C001 CMP T.DRIVE,#1
2645 023012 001034 BNE 12$
2646 023014 023737 002346 002676 CMP TMPO,HALMAX
2647 023022 101007 BHI 3$
2648 023024 052777 000004 157222 BIS #SIGN,@RLDA
2649 023032 063737 002674 002354 ADD QUAMAX,GDDAT
2650 023040 000403 BR 4$
2651 023042 163737 002674 002354 3$: SUB QUAMAX,GDDAT
2652 023050 053777 002674 157176 4$: BIS QUAMAX,@RLDA
2653 023056 012737 000001 002350 MOV #MK,TMP1
2654 023064 032777 000020 157162 BIT #DAHS,@RLDA
2655 023072 001037 BNE 5$
2656 023074 052737 000020 002350 BIS #DAHS,TMP1
2657 023102 000433 BR 5$
2658 023104 023737 002346 002726 12$: CMP TMPO,HMAX
2659 023112 101007 BHI 13$
2660 023114 052777 000004 157132 BIS #SIGN,@RLDA
2661 023122 063737 002724 002354 ADD QMAX,GDDAT
```

2662	023130	000403				BR	14\$	
2663	023132	163737	002724	002354	13\$:	SUB	QMAX,GDDAT	
2664	023140	053777	002724	157106	14\$:	BIS	QMAX,@RLDA	
2665	023146	012737	000001	002350		MOV	#MK,TMP1	
2666	023154	032777	000020	157072		BIT	#DAHS,@RLDA	
2667	023162	001003				BNE	5\$	
2668	023164	052737	000020	002350		BIS	#DAHS,TMP1	
2669	023172	004537	013414		5\$:	JSR	R5,LDFUNC	;ISSUE FUNCTION OF FOLLOWING WORD
2670	023176	000006				SEEK		;SEEK
2671	023200	004537	014246			JSR	R5,WTCRDY	;WAIT FOR CONTROLLER READY HIGH
2672	023204					CKLOOP		;CHECK IF /FL:LOE IS SET
(3)	023204	104406				TRAP	C\$CLP1	
2673								
2674								
2675	023206	004537	013114			JSR	R5,CHERR	;CHECK CONTROLLER FOR ERRORS
2676	023212					CKLOOP		;CHECK IF /FL:LOE IS SET
(3)	023212	104406				TRAP	C\$CLP1	
2677								
2678	023214	013777	002350	157032		MOV	TMP1,@RLDA	;SEND IN NEW DIFFERENCE WORD
2679	023222	004537	014246			JSR	R5,WTCRDY	;WAIT FOR CONTROLLER READY HIGH
2680	023226					CKLOOP		;CHECK IF /FL:LOE IS SET
(3)	023226	104406				TRAP	C\$CLP1	
2681								
2682	023230	004537	013114			JSR	R5,CHERR	;CHECK CONTROLLER FOR ERRORS
2683	023234					CKLOOP		;CHECK IF /FL:LOE IS SET
(3)	023234	104406				TRAP	C\$CLP1	
2684								
2685	023236	004537	014160			JSR	R5,WTCRDY	;WAIT FOR DRIVE READY
2686	023242				8\$:	CKLOOP		;CHECK IF /FL:LOE IS SET
(3)	023242	104406				TRAP	C\$CLP1	
2687								
2688								
2689	023244	004537	013114			JSR	R5,CHERR	;CHECK CONTROLLER FOR ERRORS
2690	023250					CKLOOP		;CHECK IF /FL:LOE IS SET
(3)	023250	104406				TRAP	C\$CLP1	
2691								
2692	023252	004537	013414			JSR	R5,LDFUNC	;ISSUE FUNCTION OF FOLLOWING WORD
2693	023256	000010				RDHDR		;READ HEADER
2694	023260	004537	014246			JSR	R5,WTCRDY	;WAIT FOR CONTROLLER READY HIGH
2695	023264					CKLOOP		;CHECK IF /FL:LOE IS SET
(3)	023264	104406				TRAP	C\$CLP1	
2696								
2697	023266	004537	013114			JSR	R5,CHERR	;CHECK CONTROLLER FOR ERRORS
2698	023272					ESCAPE	TST	;IF /FL:LOE SET LOOP, ELSE EXIT TST
(3)	023272	104410				TRAP	C\$ESCAPE	
(3)	023274	000036				.WORD	L10072-	
2699								
2700	023276	013737	002310	002356		MOV	E.MP,BDDAT	;READ HEADER
2701	023304	043737	002326	002356		BIC	SECMSK,BDDAT	;CLEAR SECTOR ADDRESS
2702	023312	023737	002354	002356		CMP	GDDAT,BDDAT	;IS HEADER CORRECT?
2703	023320	001404				BEQ	10\$;IF SO BRANCH
2704								
2705	023322					ERRDF	50.,EM57,ERR4	
(4)	023322	104455				TRAP	C\$ERDF	
(5)	023324	000062				.WORD	50	
(5)	023326	006551				.WORD	EM57	

CZRLGBO RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 2-36
TEST 42 - TEST TIME AT WHICH DIF WD GETS TRANSMITTED

SEQ 0099

(5) 023330 010316
2706 023332
2707
2708 023332
(3) 023332
(3) 023332 104401

.WORD ERR4
10\$:
ENDTST ;****END OF TEST****
L10072:
TRAP C\$ETST

2709
2710
2711
2712
2713 023334
2714 023334

.SBTTL **TEST 43** - EXTENSIVE CHECK OF HEADER CRC
BGNTST ;****START OF TEST****
STARS

(2)
2715
2716
2717
2718
2719 023334

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

(2)
2720
2721
2722 023334 012703 002620
2723 023340

BGNSEG MOV #SKLST,R3 ;GET LIST OF DIFFERENCE WORDS
TRAP C\$BSEG ;****START OF SEGMENT****

(3) 023340 104404
2724 023342
2725 023342 004537 013414
2726 023346 000010
2727 023350 004537 014246
2728 023354
(3) 023354 104406

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

2729
2730 023356 004537 013114
2731 023362
(3) 023362 104406

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

2732
2733 023364 013737 002310 002356
2734 023372 043737 002326 002356
2735 023400 001461
2736

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

2737
2738
2739

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

2740 023402 042737 000100 002356
2741 023410 013777 002356 156636
2742 023416 052777 000001 156630
2743 023424 004537 013414

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

2744 023430 000006
2745 023432 004537 014246
2746 023436
(3) 023436 104406

SEEK ;SEEK
JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY HIGH
CKLOOP ;CHECK IF /FL:LOE IS SET
TRAP C\$CLP1

2747
2748 023440 004537 013114
2749 023444
(3) 023444 104406

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

2750
2751 023446 004537 014160

JSR R5,WTCRDY ;WAIT FOR DRIVE READY

2752	023452			89\$:	CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023452	104406			TRAP	C\$CLP1			
2753									
2754	023454	004537	013114		JSR	R5,CHERR			;CHECK CONTROLLER FOR ERRORS
2755	023460				CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023460	104406			TRAP	C\$CLP1			
2756									
2757									
2758	023462	004537	013414		JSR	R5,LDFUNC			;ISSUE FUNCTION OF FOLLOWING WORD
2759	023466	000010			RDHDR				;READ HEADER
2760	023470	004537	014246		JSR	R5,WTCRDY			;WAIT FOR CONTROLLER READY HIGH
2761	023474			96\$:	CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023474	104406			TRAP	C\$CLP1			
2762									
2763	023476	004537	013114		JSR	R5,CHERR			;CHECK CONTROLLER FOR ERRORS
2764	023502				CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023502	104406			TRAP	C\$CLP1			
2765									
2766	023504	005037	002354		CLR	GDDAT			;CLEAR EXPECTED
2767	023510	013737	002356	002370	MOV	BDDAT,DWORD			;SAVE DIFFERENCE WORD
2768	023516	013737	002310	002356	MOV	E.MP,BDDAT			;READ HEADER
2769	023524	043737	002326	002356	BIC	SECMSK,BDDAT			;MASK OUT SECTOR BITS
2770	023532	001404			BEQ	5\$;BRANCH IF ON ZERO TRACK
2771									
2772	023534				ERRDF	51,EM54,ERR3			
(4)	023534	104455			TRAP	C\$ERDF			
(5)	023536	000063			.WORD	51			
(5)	023540	006420			.WORD	EM54			
(5)	023542	010244			.WORD	ERR3			
2773	023544			5\$:	CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023544	104406			TRAP	C\$CLP1			
2774									
2775	023546	011377	156502		MOV	(R3),@RLDA			;GET DIFFERENCE WORD
2776	023552	052777	000005	156474	BIS	#SIGN!MK,@RLDA			;SET SIGN (TOWARDS SPINDLE) AND MARKER
2777	023560	004537	013414		JSR	R5,LDFUNC			;ISSUE FUNCTION OF FOLLOWING WORD
2778	023564	000006			SEEK				;SEEK
2779	023566	004537	014246		JSR	R5,WTCRDY			;WAIT FOR CONTROLLER READY HIGH
2780	023572				CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023572	104406			TRAP	C\$CLP1			
2781									
2782	023574	004537	013114		JSR	R5,CHERR			;CHECK CONTROLLER FOR ERRORS
2783	023600				CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023600	104406			TRAP	C\$CLP1			
2784									
2785	023602	004537	014160		JSR	R5,WTRDY			;WAIT FOR DRIVE READY
2786	023606				CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023606	104406			TRAP	C\$CLP1			
2787									
2788									
2789	023610	004537	013114		JSR	R5,CHERR			;CHECK CONTROLLER FOR ERRORS
2790	023614				CKLOOP				;CHECK IF /FL:LOE IS SET
(3)	023614	104406			TRAP	C\$CLP1			
2791									
2792	023616	004537	013414		JSR	R5,LDFUNC			;ISSUE FUNCTION OF FOLLOWING WORD
2793	023622	000010			RDHDR				;READ HEADER
2794	023624	004537	014246		JSR	R5,WTCRDY			;WAIT FOR CONTROLLER READY HIGH

CZRLGB0 RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 2-38
TEST 43 - EXTENSIVE CHECK OF HEADER CRC

K 8

SEQ 0101

```
2795 023630          CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 023630 104406 TRAP          C$CLP1
2796
2797
2798 023632 004537 013114 JSR          R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
2799 023636          CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 023636 104406 TRAP          C$CLP1
2800
2801 023640 011337 002354 MOV          (R3),GDDAT          ;GET EXPECTED CYLINDER
2802 023644 011337 002370 MOV          (R3),DWORD          ;SET UP DIFFERENCE FOR SEEK
2803 023650 013737 002310 002356 MOV          E.MP,BDDAT          ;READ HEADER FROM RLMP
2804 023656 043737 002326 002356 BIC          SECMSK,BDDAT          ;CLEAR OUT SECTOR BITS
2805 023664 023737 002354 002356 CMP          GDDAT,BDDAT          ;DID SEEK GO TO THE RIGHT
2806 023672 001404          BEQ          9$              ;TRACK, IF SO, GO GET NEXT
2807
2808 023674          ERRDF          52.,EM54,ERR3
      (4) 023674 104455 TRAP          C$ERDF
      (5) 023676 000064 .WORD          52
      (5) 023700 006420 .WORD          EM54
      (5) 023702 010244 .WORD          ERR3
2809 023704          9$: CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 023704 104406 TRAP          C$CLP1
2810
2811 023706 013737 002310 023722 MOV          E.MP,10$           ;GET HEADER WORD
2812 023714 004537 013772 JSR          R5,SIMBCC          ;GO CALCULATE HEADER CRC
2813 023720 000020          16.              ;16 BITS
2814 023722 000000          10$: .WORD          0           ;HEADER GOES HERE
2815 023724 000000          .WORD          0           ;START WITH ZERO CRC
2816 023726 013737 002336 023752 MOV          CALBCC,20$
2817 023734 013737 002312 023750 MOV          E.MP1,21$
2818 023742 004537 013772 JSR          R5,SIMBCC
2819 023746 000020          16.
2820 023750 000000          21$: .WORD          0
2821 023752 000000          20$: .WORD          0
2822 023754 013737 002336 002354 MOV          CALBCC,GDDAT          ;MOVE CALCULATED CRC TO GDDAT
2823 023762 013737 002314 002356 MOV          E.MP2,BDDAT          ;GET HEADER CRC FROM RLMP
2824 023770 023737 002354 002356 CMP          GDDAT,BDDAT          ;IS CRC CORRECT?
2825 023776 001404          BEQ          11$           ;IF SO CONTINUE
2826
2827 024000          ERRDF          53.,EM42,ERR4
      (4) 024000 104455 TRAP          C$ERDF
      (5) 024002 000065 .WORD          53
      (5) 024004 006041 .WORD          EM42
      (5) 024006 010316 .WORD          ERR4
2828 024010          11$: CKLOOP          ;CHECK IF /FL:LOE IS SET
      (3) 024010 104406 TRAP          C$CLP1
2829
2830
2831 024012 005723          TST          (R3)+          ;BUMP PATTERN
2832 024014 023727 002400 000001 CMP          T.DRIVE,#1
2833 024022 001005          BNE          2$
2834 024024 020327 002720 CMP          R3,#SKEND
2835 024030 001407          BEQ          12$
2836 024032 000137 023342 JMP          1$
2837 024036 020327 002762 2$: CMP          R3,#SKEEND
2838 024042 001402          BEQ          12$
```

CZRLGBO RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 2-39
TEST 43 - EXTENSIVE CHECK OF HEADER CRC

SEQ 0102

```

2839 024044 000137 023342          JMP      1$
2840 024050          12$:
2841
2842 024050          ENDSEG          ;****END OF SEGMENT****
(3) 024050          10000$:
(3) 024050 104405          TRAP      C$ESEG
2843 024052          ENDTST          ;****END OF TEST****
(3) 024052          L10073:
(3) 024052 104401          TRAP      C$ETST
2844
2845
2846          .SBTTL  **TEST 44** - VERIFY GET STATUS WHILE DRDY IS LOW
2847
2848 024054          BGNTST          ;****START OF TEST****
2849
2850 024054          STARS
(2)          ;:*****
2851          ;:VERIFY THAT WE CAN ISSUE GET STATUS AND RECIEVE
2852          ;:THE STATUS WORD WHILE THE DRIVE IS IN NOTION SEEKING
2853 024054          STARS
(2)          ;:*****
2854
2855
2856 024054          1$:
2857 024054 004537 013414          JSR      R5,LDFUNC          ;ISSUE FUNCTION OF FOLLOWING WORD
2858 024060 000010          RDHDR          ;READ HEADER
2859 024062 004537 014246          JSR      R5,WTCRDY          ;WAIT FOR CONTROLLER READY HIGH
2860 024066          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 024066 104406          TRAP      C$CLP1
2861
2862 024070 004537 013114          JSR      R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
2863 024074          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 024074 104406          TRAP      C$CLP1
2864
2865 024076 013737 002310 002356          MOV      E.MP,BDDAT          ;READ HEADER
2866 024104 043737 002326 002356          BIC      SECMSK,BDDAT          ;CLEAR OUT SECTOR
2867 024112 001461          BEQ      5$          ;IF ON TRACK ZERO, H.S. ZERO, OK
2868
2869          ;NOT ON TRACK ZERO CALCULATE DIFFERENCE WORD AND PUT IT BACK
2870          ;ON ZERO.
2871
2872 024114 042737 000100 002356          BIC      #RHHS,BDDAT          ;CLEAR OUT HEAD SELECT
2873 024122 013777 002356 156124          MOV      BDDAT,@RLDA          ;PUT CYLINDER AS DIFFERENCE WORD
2874 024130 052777 000001 156116          BIS      #MK,@RLDA          ;SET MARKER BIT
2875 024136 004537 013414          JSR      R5,LDFUNC          ;ISSUE FUNCTION OF FOLLOWING WORD
2876 024142 000006          SEEK          ;SEEK
2877 024144 004537 014246          JSR      R5,WTCRDY          ;WAIT FOR CONTROLLER READY HIGH
2878 024150          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 024150 104406          TRAP      C$CLP1
2879
2880 024152 004537 013114          JSR      R5,CHERR          ;CHECK CONTROLLER FOR ERRORS
2881 024156          CKLOOP          ;CHECK IF /FL:LOE IS SET
(3) 024156 104406          TRAP      C$CLP1
2882
2883 024160 004537 014160          JSR      R5,WTDY          ;WAIT FOR DRIVE READY
2884 024164          CKLOOP          ;CHECK IF /FL:LOE IS SET

```

(3)	024164	104406			TRAP	C\$CLP1	
2885							
2886	024166	004537	013114		JSR	R5,CHERR	:CHECK CONTROLLER FOR ERRORS
2887	024172				CKLOOP		:CHECK IF /FL:LOE IS SET
(3)	024172	104406			TRAP	C\$CLP1	
2888							
2889							
2890	024174	004537	013414		JSR	R5,LDFUNC	:ISSUE FUNCTION OF FOLLOWING WORD
2891	024200	000010			RDHDR		:READ HEADER
2892	024202	004537	014246		JSR	R5,WTCRDY	:WAIT FOR CONTROLLER READY HIGH
2893	024206				CKLOOP		:CHECK IF /FL:LOE IS SET
(3)	024206	104406			TRAP	C\$CLP1	
2894							
2895	024210	004537	013114		JSR	R5,CHERR	:CHECK CONTROLLER FOR ERRORS
2896	024214				CKLOOP		:CHECK IF /FL:LOE IS SET
(3)	024214	104406			TRAP	C\$CLP1	
2897							
2898	024216	005037	002354		CLR	GDDAT	:CLEAR EXPECTED
2899	024222	013737	002356	002370	MOV	BDDAT,DWORD	:SAVE DIFFERENCE WORD
2900	024230	013737	002310	002356	MOV	E.MP,BDDAT	:READ HEADER
2901	024236	043737	002326	002356	BIC	SECMSK,BDDAT	:MASK OUT SECTOR BITS
2902	024244	001404			BEQ	5\$:BRANCH IF ON ZERO TRACK
2903							
2904	024246				ERRDF	54.,EM54.,ERR3	
(4)	024246	104455			TRAP	C\$ERDF	
(5)	024250	000066			.WORD	54	
(5)	024252	006420			.WORD	EM54	
(5)	024254	010244			.WORD	ERR3	
2905	024256				CKLOOP		:CHECK IF /FL:LOE IS SET
(3)	024256	104406		5\$:	TRAP	C\$CLP1	
2906							
2907	024260	012777	077601	155766	MOV	#77601,@RLDA	:GET DIFFERENCE WORD
2908	024266	052777	000005	155760	BIS	#SIGN!MK,@RLDA	:SET SIGN (TOWARDS SPINDLE) AND MARKER
2909	024274	004537	013414		JSR	R5,LDFUNC	:ISSUE FUNCTION OF FOLLOWING WORD
2910	024300	000006			SEEK		:SEEK
2911	024302	004537	014246		JSR	R5,WTCRDY	:WAIT FOR CONTROLLER READY HIGH
2912	024306				CKLOOP		:CHECK IF /FL:LOE IS SET
(3)	024306	104406			TRAP	C\$CLP1	
2913							
2914							
2915	024310	004537	013114		JSR	R5,CHERR	:CHECK CONTROLLER FOR ERRORS
2916	024314				CKLOOP		:CHECK IF /FL:LOE IS SET
(3)	024314	104406			TRAP	C\$CLP1	
2917	024316	012777	000003	155730	MOV	#MK!GSBIT,@RLDA	
2918	024324	004537	013414		JSR	R5,LDFUNC	:ISSUE FUNCTION OF FOLLOWING WORD
2919	024330	000004			GSTAT		
2920	024332	004537	014246		JSR	R5,WTCRDY	:WAIT FOR CONTROLLER READY HIGH
2921	024336				CKLOOP		:CHECK IF /FL:LOE IS SET
(3)	024336	104406			TRAP	C\$CLP1	
2922	024340	004537	013114		JSR	R5,CHERR	:CHECK CONTROLLER FOR ERRORS
2923							
2924	024344				ENDTST		:****END OF TEST****
(3)	024344				L10074:		
(3)	024344	104401			TRAP	C\$ETST	
2925							
2926	024346				BGNMOD	HRDPRM	


```

2927
2928 024346          BGNHRD
(3) 024346 000030  .WORD L10075-L$HARD/2
2929
2930 024350          GPRML  CNTMSG,CNT,1,YES
(4) 024350 005130  .WORD  T$CODE
(4) 024352 024444  .WORD  CNTMSG
(4) 024354 000001  .WORD  1
2931 024356          GPRMA  CSRMSG,CSR,0,160000,177776,YES
(4) 024356 000031  .WORD  T$CODE
(4) 024360 024430  .WORD  CSRMSG
(4) 024362 160000  .WORD  T$LLOLIM
(4) 024364 177776  .WORD  T$HILIM
2932 024366          GPRMA  VECMSG,VECT,0,0,776,YES
(4) 024366 001031  .WORD  T$CODE
(4) 024370 024462  .WORD  VECMSG
(4) 024372 000000  .WORD  T$LLOLIM
(4) 024374 000776  .WORD  T$HILIM
2933 024376          GPRMD  BRMSG,PRIOR,0,340,0,7,YES
(4) 024376 002032  .WORD  T$CODE
(4) 024400 024451  .WORD  BRMSG
(4) 024402 000340  .WORD  340
(4) 024404 000000  .WORD  T$LLOLIM
(4) 024406 000007  .WORD  T$HILIM
2934 024410          GPRML  DRTYPE,TYPDR,1,YES
(4) 024410 003130  .WORD  T$CODE
(4) 024412 024471  .WORD  DRTYPE
(4) 024414 000001  .WORD  1
2935 024416          GPRMD  DRMSG,DRBT,0,03400,0,7,YES
(4) 024416 004032  .WORD  T$CODE
(4) 024420 024513  .WORD  DRMSG
(4) 024422 003400  .WORD  03400
(4) 024424 000000  .WORD  T$LLOLIM
(4) 024426 000007  .WORD  T$HILIM
2936
2937 024430          ENDHRD
(2)                                     .EVEN
(3) 024430          L10075:
2938
2939 024430 052502 020123 042101 CSRMSG: .ASCIZ /BUS ADDRESS/
      024436 051104 051505 000123
2940 024444 046122 030461 000      CNTMSG: .ASCIZ /RL11/
2941 024451 102 020122 042514 BRMSG: .ASCIZ /BR LEVEL/
      024456 042526 000114
2942 024462 042526 052103 051117 VECMSG: .ASCIZ /VECTOR/
      024470 000
2943 024471 104 044522 042526 DRTYPE: .ASCIZ /DRIVE TYPE = RL01/
      024476 052040 050131 020105
      024504 020075 046122 030460
      024512 000
2944 024513 104 044522 042526 DRMSG: .ASCIZ /DRIVE/
      024520 000
2945 024522          .EVEN
2946
2947 024522          ENDMOD
2948

```

```
2949
2950
2951 024522          BGNMOD  SFTPRM
2952
2953 024522          BGNSFT
(3) 024522 000011   .WORD  L10076-L$SOFT/2
2954 024524          GPRML  DMSG,DLT,1,YES
(4) 024524 000130   .WORD  T$CODE
(4) 024526 024546   .WORD  DMSG
(4) 024530 000001   .WORD  1
2955 024532          XFERF  1$
(5) 024532 006044   .WORD  T$CODE
2956 024534          GPRMD  EMSG,ELI,0,177777,0,177777,YES
(4) 024534 001032   .WORD  T$CODE
(4) 024536 024572   .WORD  EMSG
(4) 024540 177777   .WORD  177777
(4) 024542 000000   .WORD  T$LOLIM
(4) 024544 177777   .WORD  T$HILIM
2957 024546          1$:    ENDSFT
(2)
(3) 024546          L10076:
2958
2962
2963 024546 051104 050117 047440 DMSG:  .ASCIZ /DROP ON ERROR LIMIT/
2964 024572 051105 047522 020122 EMSG:  .ASCIZ /ERROR LIMIT/
2965
2969
2970          .EVEN
2971
2972 024606          LASTAD  ENDMOD
2973 024606
(2)
(4) 024606 000000   .EVEN
(4) 024610 000000   .WORD  0
(3) 024612          L$LAST:: .WORD  0
2974
2975          .END
```


CZRLGBO RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 3-5
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0111

FRMT99 011207
F\$AU = 000015
F\$AUTO= 000020
F\$BGN = 000040

450	486#													
7#	711	715												
7#	641	670												
7#	15	19	24	72	74	295	298	388	393	395	403	410		
418	427	436	443	454	504	506	518	520	530	532	536	542		
549	551	638	641	672	674	695	699	701	707	709	711	717		
725	974	1013	1019	1039	1044	1065	1070	1090	1095	1115	1120	1134		
1149	1157	1162	1173	1185	1194	1199	1209	1219	1227	1232	1242	1257		
1265	1270	1280	1295	1301	1306	1317	1330	1336	1341	1351	1362	1368		
1373	1383	1393	1399	1404	1414	1425	1431	1436	1467	1472	1493	1498		
1516	1521	1558	1563	1600	1605	1645	1649	1698	1702	1726	1731	1752		
1780	1785	1817	1822	1836	1845	1848	1858	1869	1873	1893	1898	1926		
1931	1956	1961	1991	1995	2007	2011	2032	2037	2051	2054	2063	2066		
2066	2080	2085	2098	2101	2122	2127	2196	2201	2220	2225	2250	2255		
2296	2301	2319	2393	2419	2424	2457	2467	2472	2503	2514	2519	2551		
2561	2566	2596	2599	2609	2614	2698	2708	2713	2723	2843	2848	2924		
2926	2928	2947	2951	2953	2972									

F\$CLEA- 000007
F\$DU = 000016
F\$END = 000041

7#	674	693												
7#	701	705												
7#	15	19	24	72	74	295	298	388	393	401	408	416		
425	434	441	452	461	504	506	518	520	530	532	536	549		
636	638	670	672	693	695	699	705	707	709	715	717	725		
978	1013	1019	1039	1044	1065	1070	1090	1095	1115	1120	1149	1156		
1157	1162	1185	1193	1194	1199	1219	1226	1227	1232	1257	1264	1265		
1270	1295	1300	1301	1306	1330	1335	1336	1341	1362	1367	1368	1373		
1393	1398	1399	1404	1425	1430	1431	1436	1467	1472	1493	1498	1516		
1521	1558	1563	1600	1605	1645	1649	1698	1702	1726	1731	1752	1780		
1785	1817	1822	1845	1848	1858	1864	1869	1873	1893	1898	1926	1931		
1956	1961	1991	1995	2007	2011	2032	2037	2051	2054	2063	2066	2079		
2080	2085	2098	2101	2122	2127	2196	2201	2220	2225	2250	2255	2296		
2301	2393	2418	2419	2424	2457	2467	2472	2503	2514	2519	2551	2561		
2566	2596	2599	2609	2614	2698	2708	2713	2842	2843	2848	2924	2926		
2937	2947	2951	2957	2972										

F\$HARD= 000004
F\$HW = 000013
F\$INIT= 000006
F\$JMP = 000050
F\$MOD = 000000

7#	2928	2937	2955											
7#	508	516												
7#	551	636												

F\$MSG = 000011

7#	15	19	24	72	74	295	298	388	393	504	506	518		
520	530	532	536	549	638	672	695	699	707	709	717	725		
1013	2926	2947	2951	2972	410	416	418	425	427	434	436	441		
7#	395	401	403	408										
443	452	454	461											
7#	542	546												

F\$PROT= 000021
F\$PWR = 000017
F\$RPT = 000012
F\$SEG = 000003

7#	1134	1156	1173	1193	1209	1226	1242	1264	1280	1300	1317	1335		
1351	1367	1383	1398	1414	1430	1836	1864	2058	2079	2319	2418	2723		
2842														

F\$SOFT= 000005
F\$SRV = 000010
F\$SUB = 000002
F\$SW = 000014
F\$TEST= 000001

7#	2953	2955	2957											
7#	974	978												
7#	522	528												
7#	1019	1039	1044	1065	1070	1090	1095	1115	1120	1157	1162	1194		
1199	1227	1232	1265	1270	1301	1306	1336	1341	1368	1373	1399	1404		
1431	1436	1467	1472	1493	1498	1516	1521	1558	1563	1600	1605	1645		
1649	1698	1702	1726	1731	1780	1785	1817	1822	1869	1873	1893	1898		

CZRLGBO RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 3-6
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0112

		1926	1931	1956	1961	1991	1995	2007	2011	2032	2037	2080	2085	2122
		2127	2196	2201	2220	2225	2250	2255	2296	2301	2419	2424	2467	2472
		2514	2519	2561	2566	2609	2614	2708	2713	2843	2848	2924		
GDDAT	002354	115#	413	422	431	1035*	1061*	1087*	1111*	1137*	1138*	1139	1142*	1145
		1175*	1178*	1179	1181	1211*	1212	1215	1245*	1246*	1250*	1253	1283*	1284*
		1288*	1291	1320*	1323*	1326	1354*	1355*	1358	1386*	1389	1417*	1418*	1421
		1452*	1455*	1462	1485*	1508*	1542*	1551*	1573*	1576*	1577	1585	1595*	1617*
		1620*	1621	1629	1638*	1660*	1663*	1664	1673	1682*	1691*	1744*	1755	1762*
		1764	1771*	1773	1834*	1850	2056*	2057*	2070	2114*	2116	2171*	2173*	2174
		2178	2182*	2361*	2395*	2399	2434*	2461	2507*	2508	2529*	2555	2577*	2603
		2636*	2637*	2639	2642	2649*	2651*	2661*	2663*	2702	2766*	2801*	2805	2822*
		2824	2898*											
GLBDAT	002242	74#												
GLBEQA	002242	24#												
GLBERR	010152	393#												
GLBSUB	013030	725#												
GLBXTX	003766	298#												
GODRVR=	000202	48#												
G\$BIT =	000002	50#	1885	1904	1916	2544	2592	2917						
G\$STAT =	000004	43#	769	839	1887	1906	1918	1944	1974	2546	2594	2919		
G\$TINT	004527	324#	876											
G\$TMES	004470	323#	875											
G\$CNTO=	000200	7#												
G\$DELM=	000372	7#	614	985	999	2214								
G\$DISP=	000003	7#												
G\$EXCP=	000400	7#												
G\$HILI=	000002	7#												
G\$LOLI=	000001	7#												
G\$NO =	000000	7#												
G\$OFFS=	000400	7#	2930	2931	2932	2933	2934	2935	2954	2956				
G\$OF\$SI=	000376	7#	2930	2931	2932	2933	2934	2935	2954	2956				
G\$PRMA=	000001	7#	2931	2932										
G\$PRMD=	000002	7#	2933	2935	2956									
G\$PRML=	000000	7#	2930	2934	2954									
G\$RADA=	000140	7#												
G\$RADB=	000000	7#												
G\$RADD=	000040	7#												
G\$RADL=	000120	7#	2930	2934	2954									
G\$RADO=	000020	7#	2931	2932	2933	2935	2956							
G\$XFER=	000004	7#	2955											
G\$YES =	000010	7#	2930	2931	2932	2933	2934	2935	2954	2956				
HALMAX	002676	229#	2646											
HCRCME	004005	307#	801											
HDRBUF	003266	294#	2140	2154										
HDRLST	013634	850	871#											
HMAX	002726	241#	2658											
HNFMES	004013	308#	805											
HOE =	100000	26#												
HPTCOD	011620	506#												
HRDPRM	024346	2926#												
IBE =	010000	26#												
IDU =	000040	26#												
IER =	020000	26#												
INITCO	012006	549#												
INTEN =	000100	28#	681	848	856	1806	1842	1906	1974	2022	2271			
INTFLG	002322	102#	976*	1803*	1808	1811*	1838*	1853	1860	1902*	1909	1914*	1921	1971*

CZRLGBO RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 3-8
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0114

L\$AUT	002070	G	17#				
L\$AUTO	012530	G	17	641#			
L\$CCP	002106	G	17#				
L\$CLEA	012752	G	17	674#			
L\$CO	002032	G	17#				
L\$DEPO	002011	G	17#				
L\$DESC	002122	G	17	21#			
L\$DESP	002076	G	17#				
L\$DEVP	002060	G	17#				
L\$DISP	011650	G	17	534#			
L\$DLY	002116	G	17#	614	985	999	2214
L\$DTP	002040	G	17#				
L\$DTYP	002034	G	17#				
L\$DU	013020	G	17	701#			
L\$DUT	002072	G	17#				
L\$DVTY	002230	G	17	22#			
L\$EF	002052	G	17#				
L\$ENVI	002044	G	17#				
L\$ETP	002102	G	17#				
L\$EXP1	002046	G	17#				
L\$EXP4	002064	G	17#				
L\$EXP5	002066	G	17#				
L\$HARD	024350	G	17	2928#			
L\$HIME	002120	G	17#				
L\$HPCP	002016	G	17#				
L\$HPTP	002022	G	17#				
L\$HW	011622	G	17	508#			
L\$ICP	002104	G	17#				
L\$INIT	012006	G	17	551#			
L\$LADP	002026	G	17#				
L\$LAST	024612	G	17	2973#			
L\$LOAD	002100	G	17#				
L\$LUN	002074	G	17#				
L\$MREV	002050	G	17#				
L\$NAME	002000	G	17#				
L\$PRIO	002042	G	17#				
L\$PROT	012000	G	17	542#			
L\$PRT	002112	G	17#				
L\$REPP	002062	G	17#				
L\$REV	002010	G	17#				
L\$SOFT	024524	G	17	2953#			
L\$SPC	002056	G	17#				
L\$SPCP	002020	G	17#				
L\$SPTP	002024	G	17#				
L\$STA	002030	G	17#				
L\$SW	011640	G	17	522#			
L\$TEST	002114	G	17#				
L\$TIML	002014	G	17#				
L\$UNIT	002012	G	17#	556	575		
L10000	010166		401#				
L10001	010200		408#				
L10002	010242		416#				
L10003	010314		425#				
L10004	010362		434#				
L10005	010374		441#				
L10006	010436		452#				

CZRLGB0 RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 3-9
CROSS REFERENCE TABLE -- USER SYMBOLS

L10007	010474	461#		
L10010	011636	508	516#	
L10011	011646	522	528#	
L10013	012526	636#		
L10014	012750	670#		
L10015	013016	693#		
L10016	013022	705#		
L10017	013026	715#		
L10020	014156	978#		
L10021	014440	1039#		
L10022	014534	1065#		
L10023	014630	1090#		
L10024	014724	1115#		
L10025	015044	1157#		
L10026	015146	1194#		
L10027	015234	1227#		
L10030	015360	1265#		
L10031	015504	1301#		
L10032	015610	1336#		
L10033	015710	1368#		
L10034	016000	1399#		
L10035	016100	1431#		
L10036	016210	1467#		
L10037	016262	1493#		
L10040	016320	1516#		
L10041	016444	1558#		
L10042	016604	1600#		
L10043	016744	1645#		
L10044	017150	1698#		
L10045	017200	1726#		
L10046	017404	1752	1780#	
L10047	017470	1817#		
L10050	017634	1845	1848	1869#
L10051	017664	1893#		
L10052	020036	1926#		
L10053	020124	1956#		
L10054	020252	1991#		
L10055	020274	2007#		
L10056	020354	2032#		
L10057	020520	2051	2054	2080#
L10060	020656	2098	2101	2122#
L10061	021174	2196#		
L10062	021270	2220#		
L10063	021334	2250#		
L10064	021460	2296#		
L10065	022076	2419#		
L10066	022230	2457	2467#	
L10067	022372	2503	2514#	
L10070	022532	2551	2561#	
L10071	022704	2596	2599	2609#
L10072	023332	2698	2708#	
L10073	024052	2843#		
L10074	024344	2924#		
L10075	024430	2928	2937#	
L10076	024546	2953	2957#	
MAXCYL	002372	122#		

MAXSEC	002366	120#													
MDHEDR	002000	G 15#													
MERLMT	011642	525#	733												
MK	= 000001	51#	1885	1904	1916	1942	1972	2209	2236	2269	2338	2371	2433	2481	
		2527	2544	2575	2592	2638	2653	2665	2742	2776	2874	2908	2917		
MSCRLF	004032	311#	816												
MXSEC1	002364	119#	2180												
NOOPO	= 000000	40#	1719	1747	1806	1842									
NOOP7	= 000016	41#													
NOPINT	004151	316#	872												
NOPMES	004120	315#	871												
NOPWR	012030	555	558#												
NXM	= 020000	35#	790												
NXMMES	003773	305#	793												
NXT	012100	572#	584	586											
OKHDR	013644	875#													
OPI	= 002000	31#	794	1951	1986										
OPIERR	004045	314#	1948	1983											
OPIMES	004000	306#	797												
OSAPTS	= 000000	7#	17												
OSAU	= 000001	7#	13#	17											
OSBGNR	= 000000	7#	17												
OSBGNS	= 000001	7#	13#	17											
OSDU	= 000001	7#	13#	17											
OSERRT	= 000000	7#	17												
OSGNSW	= 000001	7#	13#	17											
OSPOIN	= 000001	7#	13#	17											
OSSETU	= 000000	7#	17	2973											
PFLG	002316	100#	631*	634*											
PNT	= 001000	G 26#													
PRI	= 002000	G 26#													
PRIOR	= 000004	61#	2933												
PRI00	= 000000	G 26#	1804	1903	1915	1970	2019	2268							
PRI01	= 000040	G 26#													
PRI02	= 000100	G 26#													
PRI03	= 000140	G 26#													
PRI04	= 000200	G 26#													
PRI05	= 000240	G 26#													
PRI06	= 000300	G 26#													
PRI07	= 000340	G 26#	676	1450	1908	1920	1976	2024	2289						
PWRFLG	002242	78#	556*	583	585*	602	687	690*							
QMAX	002724	240#	2661	2663	2664										
QUAMAX	002674	228#	2649	2651	2652										
RDHDR	= 000010	45#	2002	2022	2049	2061	2096	2142	2322	2354	2387	2452	2498	2629	
		2693	2726	2759	2793	2858	2891								
READ	= 000014	47#													
REST	012146	581#													
RESTMS	013376	818#	860*	1948*	1983*	2159*									
RHDINT	004344	320#	880												
RHDMES	004304	319#	879	2159											
RHS	= 000100	53#	2336	2507	2639	2643	2740	2872							
RLBA	002252	82#	597*	900	910	1057	1061	1179*	1180	1319*	1324*	1325	1353*	1356*	
		1357	1481*	1484*	1487	1533*	1539	1543	1579*	1622*	1635	1639	1665*	1679	
		1683	1742*	1763											
RLCS	002250	81#	463	595*	609*	610*	611	625*	626*	627*	628	651	661*	662*	
		663	678	681*	841*	843	863*	865*	899	909	983	997	1031	1035	

		1139*	1140	1143	1244*	1247*	1248	1251	1282*	1285*	1286	1289	1451*	1453
		1460	1535*	1574	1577*	1583	1618	1621*	1627	1661	1664*	1671	2144*	2145*
		2146	2148											
RLDA	002254	83#	599*	624*	838*	901	911	1082	1087	1212*	1214	1385*	1387*	1388
		1416*	1419*	1420	1507*	1510	1534*	1548	1552	1578*	1591	1596	1623*	1666*
		1687	1690	1741*	1772	1885*	1904*	1916*	1942*	1972*	2209*	2236*	2269*	2337*
		2338*	2370*	2371*	2433*	2481*	2482*	2527*	2544*	2575*	2576*	2592*	2638*	2641*
		2648*	2652*	2654	2660*	2664*	2666	2678*	2741*	2742*	2775*	2776*	2873*	2874*
		2907*	2908*	2917*										
RLMP	002256	84#	601*	902	912	913	914	1107	1111	1667*	1743*	1744	1754	2149
		2150	2151											
RL2	002722	239#												
SECMSK	002326	104#	2057	2069	2176	2182	2330	2364	2398	2637	2701	2734	2769	2804
		2866	2901											
SEEK =	000006	44#	2211	2238	2271	2340	2373	2438	2484	2531	2579	2670	2744	2778
		2876	2910											
SEKINT	004436	322#	878											
SEKMS	004405	321#	877											
SF TPRM	024522	G	2951#											
SIGN =	000004	52#	2209	2269	2371	2648	2660	2776	2908					
SIMBCC	013772	931#	2104	2110	2812	2818								
SIZE =	000004	70#												
SKEEND	002762	256#	2412	2837										
SKEND	002720	238#	2408	2834										
SKLST	002620	206#	2318	2722										
SPTCOD	011636	G	520#											
START	012106	567	574#											
START1	012050	559	562#											
STHS =	000100	54#	2577											
SVCGBL =	000000	7#	15	17	21	22	24	74	298	393	395	403	410	418
		427	436	443	454	506	508	520	522	532	534	542	549	551
		641	672	674	699	701	709	711	725	2926	2928	2951	2953	2973#
SVCINS-	000000	7#	8#	17	21	22	401	408	413	416	422	425	431	434
		441	450	452	457	461	463	466	467	468	469	472	473	508
		522	534	553	554	555	558	559	560	561	569	570	581	582
		614	619	622	623	630	632	633	636	643	652	657	659	665
		668	670	676	683	693	705	715	728	729	736	738	739	821
		978	985	989	999	1005	1029	1032	1037	1038	1039	1055	1058	1063
		1064	1065	1080	1083	1088	1089	1090	1105	1108	1113	1114	1115	1134
		1148	1149	1156	1157	1173	1184	1185	1193	1194	1209	1218	1219	1226
		1227	1242	1256	1257	1264	1265	1280	1294	1295	1300	1301	1317	1329
		1330	1335	1336	1351	1361	1362	1367	1368	1383	1392	1393	1398	1399
		1414	1424	1425	1430	1431	1450	1457	1465	1467	1486	1490	1493	1509
		1513	1516	1545	1546	1554	1558	1588	1589	1598	1600	1632	1633	1641
		1645	1676	1677	1685	1686	1693	1698	1721	1726	1749	1752	1758	1760
		1767	1769	1776	1780	1804	1810	1812	1817	1836	1839	1845	1848	1856
		1858	1864	1869	1889	1893	1903	1908	1911	1912	1915	1920	1923	1924
		1926	1950	1953	1956	1970	1976	1979	1980	1985	1988	1991	2004	2007
		2019	2024	2027	2028	2032	2051	2054	2058	2063	2066	2073	2075	2079
		2080	2098	2101	2119	2122	2189	2190	2196	2214	2217	2220	2240	2243
		2246	2250	2268	2276	2277	2281	2287	2289	2293	2294	2296	2319	2324
		2327	2342	2345	2348	2351	2356	2359	2367	2368	2375	2378	2381	2384
		2390	2393	2402	2403	2418	2419	2440	2443	2446	2449	2454	2457	2464
		2467	2486	2489	2492	2495	2500	2503	2511	2514	2533	2536	2539	2542
		2548	2551	2558	2561	2581	2584	2587	2590	2596	2599	2606	2609	2631
		2634	2672	2676	2680	2683	2686	2690	2695	2698	2705	2708	2723	2728

	2731	2746	2749	2752	2755	2761	2764	2772	2773	2780	2783	2786	2790
	2795	2799	2808	2809	2827	2828	2842	2843	2860	2863	2878	2881	2884
	2887	2893	2896	2904	2905	2912	2916	2921	2924	2928	2930	2931	2932
	2933	2934	2935	2937	2953	2954	2955	2956	2957	2973			
SVCSUB= 177777	7#												
SVCTAG= 000000	7#	9#	401	408	416	425	434	441	452	461	516	528	636
	670	693	705	715	978	1020	1025	1039	1047	1052	1065	1071	1076
	1090	1096	1101	1115	1124	1129	1156	1157	1164	1169	1193	1194	1201
	1205	1226	1227	1233	1238	1264	1265	1272	1276	1300	1301	1308	1313
	1335	1336	1343	1347	1367	1368	1375	1379	1398	1399	1406	1410	1430
	1431	1438	1447	1467	1474	1478	1493	1500	1504	1516	1523	1529	1558
	1564	1570	1600	1608	1614	1645	1652	1657	1698	1706	1711	1726	1733
	1736	1780	1787	1797	1817	1824	1828	1864	1869	1876	1882	1893	1926
	1933	1939	1956	1962	1967	1991	1996	1999	2007	2013	2016	2032	2040
	2044	2079	2080	2087	2091	2122	2130	2136	2196	2202	2206	2220	2228
	2232	2250	2258	2262	2296	2306	2315	2418	2419	2428	2431	2467	2475
	2478	2514	2521	2525	2561	2568	2572	2609	2617	2625	2708	2714	2719
	2842	2843	2850	2853	2924	2937	2957						
SVCTST= 177777	7#	1019	1044	1070	1095	1120	1162	1199	1232	1270	1306	1341	1373
	1404	1436	1472	1498	1521	1563	1605	1649	1702	1731	1785	1822	1873
	1898	1931	1961	1995	2011	2037	2085	2127	2201	2225	2255	2301	2424
	2472	2519	2566	2614	2713	2848							
SVHD 002374	123#												
SLSYM= 010000	7#	401#	408#	416#	425#	434#	441#	452#	461#	516#	528#	636#	670#
	693#	705#	715#	978#	1039#	1065#	1090#	1115#	1134#	1157#	1173#	1194#	1209#
	1227#	1242#	1265#	1280#	1301#	1317#	1336#	1351#	1368#	1383#	1399#	1414#	1431#
	1467#	1493#	1516#	1558#	1600#	1645#	1698#	1726#	1780#	1817#	1836#	1869#	1893#
	1926#	1956#	1991#	2007#	2032#	2058#	2080#	2122#	2196#	2220#	2250#	2296#	2319#
	2419#	2467#	2514#	2561#	2609#	2708#	2723#	2843#	2924#	2937#	2957#		
TEMP2 002340	109#	934*	957*	2178*	2179*	2180							
TEMP3 002342	110#	935*	939*										
TEMP4 002344	111#	936*	938	948*	950	955*	956*	959					
TMPFNC 002404	128#	769	772	849*									
TMP0 002346	112#	422	2642*	2643*	2646	2658							
TMP1 002350	113#	2653*	2656*	2665*	2668*	2678							
TMP2 002352	114#												
TRPFLG 002320	101#	642*	655	971*	1028*	1033	1054*	1059	1079*	1084	1104*	1109	
TRPHAN 014144	643	646	971#	1029	1055	1080	1105						
TYPDR = 000006	62#	2934											
T\$ARGC= 000001	17#	413#	422#	431#	450#	457#	463#	466#	467#	468#	469#	472#	473#
	619#	657#	665#	736#									
T\$CODE= 001032	2930#	2931#	2932#	2933#	2934#	2935#	2954#	2955#	2956#				
T\$ERRN= 000066	7#	821#	989#	1005#	1037#	1063#	1088#	1113#	1148#	1184#	1218#	1256#	1294#
	1329#	1361#	1392#	1424#	1465#	1490#	1513#	1545#	1554#	1588#	1598#	1632#	1641#
	1676#	1685#	1693#	1758#	1767#	1776#	1810#	1856#	1911#	1923#	1953#	1979#	1988#
	2027#	2073#	2119#	2189#	2276#	2293#	2367#	2402#	2464#	2511#	2558#	2606#	2705#
	2772#	2808#	2827#	2904#									
T\$EXCP= 000000	2931#	2932#	2933#	2935#	2956#								
T\$FLAG= 000040	1149#	1185#	1219#	1257#	1295#	1330#	1362#	1393#	1425#	1752#	1845#	1848#	1858#
	2051#	2054#	2063#	2066#	2098#	2101#	2393#	2457#	2503#	2551#	2596#	2599#	2698#
T\$GMAN= 000000	7#												
T\$HILI= 177777	2931#	2932#	2933#	2935#	2956#								
T\$LAST= 000001	7#	2973#											
T\$LOLI= 000000	2931#	2932#	2933#	2935#	2956#								
T\$LSYM= 010000	7#	401	408	416	425	434	441	452	461	516	528	636	670
	693	705	715	978	1039	1065	1090	1115	1157	1194	1227	1265	1301

	1336	1368	1399	1431	1467	1493	1516	1558	1600	1645	1698	1726	1780
	1817	1869	1893	1926	1956	1991	2007	2032	2080	2122	2196	2220	2250
	2296	2419	2467	2514	2561	2609	2708	2843	2924	2937	2957		
TSLTNO= 000054	2973#												
TSNEST= 177777	7#	15#	19#	24#	72#	74#	295#	298#	388#	393#	395#	401#	403#
	408#	410#	416#	418#	425#	427#	434#	436#	441#	443#	452#	454#	461#
	504#	506#	508#	516#	518#	520#	522#	528#	530#	532#	536#	542#	546#
	549#	551#	636#	638#	641#	670#	672#	674#	693#	695#	699#	701#	705#
	707#	709#	711#	715#	717#	725#	974#	978#	1013#	1019#	1039#	1044#	1065#
	1070#	1090#	1095#	1115#	1120#	1134#	1156#	1157#	1162#	1173#	1193#	1194#	1199#
	1209#	1226#	1227#	1232#	1242#	1264#	1265#	1270#	1280#	1300#	1301#	1306#	1317#
	1335#	1336#	1341#	1351#	1367#	1368#	1373#	1383#	1398#	1399#	1404#	1414#	1430#
	1431#	1436#	1467#	1472#	1493#	1498#	1516#	1521#	1558#	1563#	1600#	1605#	1645#
	1649#	1698#	1702#	1726#	1731#	1780#	1785#	1817#	1822#	1836#	1864#	1869#	1873#
	1893#	1898#	1926#	1931#	1956#	1961#	1991#	1995#	2007#	2011#	2032#	2037#	2058#
	2079#	2080#	2085#	2122#	2127#	2196#	2201#	2220#	2225#	2250#	2255#	2296#	2301#
	2319#	2418#	2419#	2424#	2467#	2472#	2514#	2519#	2561#	2566#	2609#	2614#	2708#
	2713#	2723#	2842#	2843#	2848#	2924#	2926#	2928#	2937#	2947#	2951#	2953#	2955
	2957#	2972#											
TSNSO = 000000	15#	19	24#	72	74#	295	298#	388	393#	504	506#	518	520#
	530	532#	536	542#	546	549#	638	641#	670	672#	695	699#	707
	709#	717	725#	1013	1019#	1039	1044#	1065	1070#	1090	1095#	1115	1120#
	1157	1162#	1194	1199#	1227	1232#	1265	1270#	1301	1306#	1336	1341#	1368
	1373#	1399	1404#	1431	1436#	1467	1472#	1493	1498#	1516	1521#	1558	1563#
	1600	1605#	1645	1649#	1698	1702#	1726	1731#	1780	1785#	1817	1822#	1869
	1873#	1893	1898#	1926	1931#	1956	1961#	1991	1995#	2007	2011#	2032	2037#
	2080	2085#	2122	2127#	2196	2201#	2220	2225#	2250	2255#	2296	2301#	2419
	2424#	2467	2472#	2514	2519#	2561	2566#	2609	2614#	2708	2713#	2843	2848#
	2924	2926#	2947	2951#	2972								
TSNS1 = 000005	395#	401	403#	408	410#	416	418#	425	427#	434	436#	441	443#
	452	454#	461	508#	516	522#	528	551#	636	674#	693	701#	705
	711#	715	974#	978	1134#	1156	1173#	1193	1209#	1226	1242#	1264	1280#
	1300	1317#	1335	1351#	1367	1383#	1398	1414#	1430	1836#	1864	2058#	2079
	2319#	2418	2723#	2842	2928#	2937	2953#	2955	2957				
TSPTNU= 000000	7#												
TSSAVL= 177777	7#												
T\$SEGL= 177777	7#	1134#	1149	1156#	1173#	1185	1193#	1209#	1219	1226#	1242#	1257	1264#
	1280#	1295	1300#	1317#	1330	1335#	1351#	1362	1367#	1383#	1393	1398#	1414#
	1425	1430#	1836#	1858	1864#	2058#	2063	2066	2079#	2319#	2393	2418#	2723#
	2842#												
TSSEKO= 010000	1134#	1149	1156	1173#	1185	1193	1209#	1219	1226	1242#	1257	1264	1280#
	1295	1300	1317#	1330	1335	1351#	1362	1367	1383#	1393	1398	1414#	1425
	1430	1836#	1858	1864	2058#	2063	2066	2079	2319#	2393	2418	2723#	2842
TS\$SUBN= 000000	7#	1019#	1044#	1070#	1095#	1120#	1162#	1199#	1232#	1270#	1306#	1341#	1373#
	1404#	1436#	1472#	1498#	1521#	1563#	1605#	1649#	1702#	1731#	1785#	1822#	1873#
	1898#	1931#	1961#	1995#	2011#	2037#	2085#	2127#	2201#	2225#	2255#	2301#	2424#
	2472#	2519#	2566#	2614#	2713#	2848#							
T\$TAGL= 177777	7#												
T\$TAGN= 010077	7#	395#	403#	410#	418#	427#	436#	443#	454#	508#	522#	542#	551#
	641#	674#	701#	711#	974#	1019#	1044#	1070#	1095#	1120#	1162#	1199#	1232#
	1270#	1306#	1341#	1373#	1404#	1436#	1472#	1498#	1521#	1563#	1605#	1649#	1702#
	1731#	1785#	1822#	1873#	1898#	1931#	1961#	1995#	2011#	2037#	2085#	2127#	2201#
	2225#	2255#	2301#	2424#	2472#	2519#	2566#	2614#	2713#	2848#	2928#	2953#	
T\$TEMP= 000000	19#	72#	295#	388#	401#	408#	416#	425#	434#	441#	452#	461#	504#
	516#	518#	528#	530#	534#	536#	546#	636#	638#	670#	693#	695#	705#
	707#	715#	717#	978#	1013#	1020#	1025#	1039#	1047#	1052#	1065#	1071#	1076#

	1090#	1096#	1101#	1115#	1124#	1129#	1149#	1156#	1157#	1164#	1169#	1185#	1193#
	1194#	1201#	1205#	1219#	1226#	1227#	1233#	1238#	1257#	1264#	1265#	1272#	1276#
	1295#	1300#	1301#	1308#	1313#	1330#	1335#	1336#	1343#	1347#	1362#	1367#	1368#
	1375#	1379#	1393#	1398#	1399#	1406#	1410#	1425#	1430#	1431#	1438#	1447#	1467#
	1474#	1478#	1493#	1500#	1504#	1516#	1523#	1529#	1558#	1564#	1570#	1600#	1608#
	1614#	1645#	1652#	1657#	1698#	1706#	1711#	1726#	1733#	1736#	1752#	1780#	1787#
	1797#	1817#	1824#	1828#	1845#	1848#	1858#	1864#	1869#	1876#	1882#	1893#	1926#
	1933#	1939#	1956#	1962#	1967#	1991#	1996#	1999#	2007#	2013#	2016#	2032#	2040#
	2044#	2051#	2054#	2063#	2066#	2079#	2080#	2087#	2091#	2098#	2101#	2122#	2130#
	2136#	2196#	2202#	2206#	2220#	2228#	2232#	2250#	2258#	2262#	2296#	2306#	2315#
	2393#	2418#	2419#	2428#	2431#	2457#	2467#	2475#	2478#	2503#	2514#	2521#	2525#
	2551#	2561#	2568#	2572#	2596#	2599#	2609#	2617#	2625#	2698#	2708#	2714#	2719#
	2842#	2843#	2850#	2853#	2924#	2930#	2931#	2932#	2933#	2934#	2935#	2937#	2947#
TSTEST= 000054	2954#	2956#	2957#	2972#									
	7#	1019#	1044#	1070#	1095#	1120#	1162#	1199#	1232#	1270#	1306#	1341#	1373#
	1404#	1436#	1472#	1498#	1521#	1563#	1605#	1649#	1702#	1731#	1785#	1822#	1873#
	1898#	1931#	1961#	1995#	2011#	2037#	2085#	2127#	2201#	2225#	2255#	2301#	2424#
TSTSTM= 177777	2472#	2519#	2566#	2614#	2713#	2848#	2973						
	7#	401	408	413	416	422	425	431	434	441	450	452	457
	461	463	466	467	468	469	472	473	553	554	558	560	569
	581	619	622	623	630	632	636	643	652	657	659	665	668
	670	676	683	693	705	715	728	736	738	739	821	989	1005
	1029	1032	1037	1038	1039	1055	1058	1063	1064	1065	1080	1083	1088
	1089	1090	1105	1108	1113	1114	1115	1134	1148	1149	1156	1157	1173
	1184	1185	1193	1194	1209	1218	1219	1226	1227	1242	1256	1257	1264
	1265	1280	1294	1295	1300	1301	1317	1329	1330	1335	1336	1351	1361
	1362	1367	1368	1383	1392	1393	1398	1399	1414	1424	1425	1430	1431
	1450	1457	1465	1467	1486	1490	1493	1509	1513	1516	1545	1546	1554
	1558	1588	1589	1598	1600	1632	1633	1641	1645	1676	1677	1685	1686
	1693	1698	1721	1726	1749	1752	1758	1760	1767	1769	1776	1780	1804
	1810	1812	1817	1836	1839	1845	1848	1856	1858	1864	1869	1889	1893
	1903	1908	1911	1912	1915	1920	1923	1924	1926	1950	1953	1956	1970
	1976	1979	1980	1985	1988	1991	2004	2007	2019	2024	2027	2028	2032
	2051	2054	2058	2063	2066	2073	2075	2079	2080	2098	2101	2119	2122
	2189	2190	2196	2217	2220	2240	2243	2246	2250	2268	2276	2277	2281
	2287	2289	2293	2294	2296	2319	2324	2327	2342	2345	2348	2351	2356
	2359	2367	2368	2375	2378	2381	2384	2390	2393	2402	2403	2418	2419
	2440	2443	2446	2449	2454	2457	2464	2467	2486	2489	2492	2495	2500
	2503	2511	2514	2533	2536	2539	2542	2548	2551	2558	2561	2581	2584
	2587	2590	2596	2599	2606	2609	2631	2634	2672	2676	2680	2683	2686
	2690	2695	2698	2705	2708	2723	2728	2731	2746	2749	2752	2755	2761
	2764	2772	2773	2780	2783	2786	2790	2795	2799	2808	2809	2827	2828
	2842	2843	2860	2863	2878	2881	2884	2887	2893	2896	2904	2905	2912
	2916	2921	2924										
TSTSTS= 000001	7#	1019#	1044#	1070#	1095#	1120#	1162#	1199#	1232#	1270#	1306#	1341#	1373#
	1404#	1436#	1472#	1498#	1521#	1563#	1605#	1649#	1702#	1731#	1785#	1822#	1873#
	1898#	1931#	1961#	1995#	2011#	2037#	2085#	2127#	2201#	2225#	2255#	2301#	2424#
	2472#	2519#	2566#	2614#	2713#	2848#							
TSSAU = 010017	711#	715											
TSSAUT= 010014	641#	670											
TSSCLE= 010015	674#	693											
TSSDU = 010016	701#	705											
TSSHAR= 010075	2928#	2937											
TSSHW = 010010	508#	516											
TSSINI= 010013	551#	636											
TSSMSG= 010007	395#	401	403#	408	410#	416	418#	425	427#	434	436#	441	443#

CZRLGBO RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 3-16
CROSS REFERENCE TABLE -- USER SYMBOLS

T43	023334 G	534	2713#											
T44	024054 G	534	2848#											
T5	014726 G	534	1120#											
T6	015046 G	534	1162#											
T7	015150 G	534	1199#											
T8	015236 G	534	1232#											
T9	015362 G	534	1270#											
UAM =	000200 G	26#												
UNITST	002246	80#	574*	578*	581	622	659	668	738					
UUT	002244	79#	572	575*	580*									
VECMG	024462	2932	2942#											
VECT =	000002	60#	2932											
WAITO	012326	614#	616											
WAIT1	021224	2214#	2216											
WCKINT	004243	318#	874											
WCKMES	004203	317#	873											
WHY	002376	124#												
WRCHK =	000002	42#	772											
WRITE =	000012	46#												
WTCRDY	014246	995#	1720	1748	1807	1844	1888	1907	1919	1945	1975	2003	2023	2050
		2062	2097	2212	2239	2272	2323	2341	2355	2374	2389	2439	2453	2485
		2499	2532	2547	2580	2595	2630	2671	2679	2694	2727	2745	2760	2779
		2794	2859	2877	2892	2911	2920							
WTDRDY	014160	981#	2245	2286	2347	2380	2445	2491	2538	2586	2685	2751	2785	2883
XPOLY	002330	105#	944	955										
XXX	012130	573	578#											
X\$ALWA=	000000	7#												
X\$FALS=	000040	7#	2955											
X\$OFFS=	000400	7#	2955											
X\$TRUE=	000020	7#												
.	- 024612	10#	293#	294#	381#	614	985	999	1149	1185	1219	1257	1295	1330
		1362	1393	1425	1752	1845	1848	1858	2051	2054	2063	2066	2098	2101
		2214	2393	2457	2503	2551	2596	2599	2698	2945#	2955			

ERRSF	1037	1063	1088	1113											
ESCAPE	1149	1185	1219	1257	1295	1330	1362	1393	1425	1752	1845	1848	1858	2051	2054
	2063	2066	2098	2101	2393	2457	2503	2551	2596	2599	2698				
GPHARD	581														
GPRMA	2931	2932													
GPRMD	2933	2935	2956												
GPRML	2930	2934	2954												
HEADER	17														
INLOOP	728														
LASTAD	2973														
MSBYTE	17#														
MSCNTO	2930#	2931#	2932#	2933#	2934#	2935#	2954#	2956#							
MSCOUN	413#	422#	431#	450#	457#	463#	466#	467#	468#	469#	472#	473#	619#	657#	665#
	736#														
MSDATA	17#	21#	22#												
MSDECR	19#	72#	295#	388#	401#	408#	416#	425#	434#	441#	452#	461#	504#	516#	518#
	528#	530#	536#	546#	636#	638#	670#	693#	695#	705#	707#	715#	717#	978#	1013#
	1039#	1065#	1090#	1115#	1156#	1157#	1193#	1194#	1226#	1227#	1264#	1265#	1300#	1301#	1335#
	1336#	1367#	1368#	1398#	1399#	1430#	1431#	1467#	1493#	1516#	1558#	1600#	1645#	1698#	1726#
	1780#	1817#	1864#	1869#	1893#	1926#	1956#	1991#	2007#	2032#	2079#	2080#	2122#	2196#	2220#
	2250#	2296#	2418#	2419#	2467#	2514#	2561#	2609#	2708#	2842#	2843#	2924#	2937#	2947#	2957#
	2972#														
MSDEFA	2930#	2931#	2932#	2933#	2934#	2935#	2954#	2956#							
MSENDE	19#	72#	295#	388#	401#	408#	416#	425#	434#	441#	452#	461#	504#	516#	518#
	528#	530#	536#	636#	638#	670#	693#	695#	705#	707#	715#	717#	978#	1013#	1039#
	1065#	1090#	1115#	1156#	1157#	1193#	1194#	1226#	1227#	1264#	1265#	1300#	1301#	1335#	1336#
	1367#	1368#	1398#	1399#	1430#	1431#	1467#	1493#	1516#	1558#	1600#	1645#	1698#	1726#	1780#
	1817#	1864#	1869#	1893#	1926#	1956#	1991#	2007#	2032#	2079#	2080#	2122#	2196#	2220#	2250#
	2296#	2418#	2419#	2467#	2514#	2561#	2609#	2708#	2842#	2843#	2924#	2937#	2947#	2957#	2972#
MSERRI	821#	989#	1005#	1037#	1063#	1088#	1113#	1148#	1184#	1218#	1256#	1294#	1329#	1361#	1392#
	1424#	1465#	1490#	1513#	1545#	1554#	1588#	1598#	1632#	1641#	1676#	1685#	1693#	1758#	1767#
	1776#	1810#	1856#	1911#	1923#	1953#	1979#	1988#	2027#	2073#	2119#	2189#	2276#	2293#	2367#
	2402#	2464#	2511#	2558#	2606#	2705#	2772#	2808#	2827#	2904#					
MSESCA	1149#	1185#	1219#	1257#	1295#	1330#	1362#	1393#	1425#	1752#	1845#	1848#	1858#	2051#	2054#
MSDESCS	2063#	2066#	2098#	2101#	2393#	2457#	2503#	2551#	2596#	2599#	2698#				
MSDESCS	1149#	1185#	1219#	1257#	1295#	1330#	1362#	1393#	1425#	1752#	1845#	1848#	1858#	2051#	2054#
	2063#	2066#	2098#	2101#	2393#	2457#	2503#	2551#	2596#	2599#	2698#				
MSEXCP	2931#	2932#	2933#	2935#	2956#										
MSGEN	15#	17#	21#	22#	24#	74#	298#	393#	395#	401#	403#	408#	410#	416#	418#
	425#	427#	434#	436#	441#	443#	452#	454#	461#	506#	508#	516#	520#	522#	528#
	532#	534#	542#	549#	551#	636#	641#	670#	672#	674#	693#	699#	701#	705#	709#
	711#	715#	725#	974#	978#	1019#	1039#	1044#	1065#	1070#	1090#	1095#	1115#	1120#	1156#
	1157#	1162#	1193#	1194#	1199#	1226#	1227#	1232#	1264#	1265#	1270#	1300#	1301#	1306#	1335#
	1336#	1341#	1367#	1368#	1373#	1398#	1399#	1404#	1430#	1431#	1436#	1467#	1472#	1493#	1498#
	1516#	1521#	1558#	1563#	1600#	1605#	1645#	1649#	1698#	1702#	1726#	1731#	1780#	1785#	1817#
	1822#	1864#	1869#	1873#	1893#	1898#	1926#	1931#	1956#	1961#	1991#	1995#	2007#	2011#	2032#
	2037#	2079#	2080#	2085#	2122#	2127#	2196#	2201#	2220#	2225#	2250#	2255#	2296#	2301#	2418#
	2419#	2424#	2467#	2472#	2514#	2519#	2561#	2566#	2609#	2614#	2708#	2713#	2842#	2843#	2848#
	2924#	2926#	2928#	2937#	2951#	2953#	2957#	2973#							
MSGETS	19#	72#	295#	388#	401#	408#	416#	425#	434#	441#	452#	461#	504#	516#	518#
	528#	530#	536#	546#	636#	638#	670#	693#	695#	705#	707#	715#	717#	978#	1013#
	1039#	1065#	1090#	1115#	1149#	1156#	1157#	1185#	1193#	1194#	1219#	1226#	1227#	1257#	1264#
	1265#	1295#	1300#	1301#	1330#	1335#	1336#	1362#	1367#	1368#	1393#	1398#	1399#	1425#	1430#
	1431#	1467#	1493#	1516#	1558#	1600#	1645#	1698#	1726#	1780#	1817#	1858#	1864#	1869#	1893#
	1926#	1956#	1991#	2007#	2032#	2063#	2066#	2079#	2080#	2122#	2196#	2220#	2250#	2296#	2393#
	2418#	2419#	2467#	2514#	2561#	2609#	2708#	2842#	2843#	2924#	2937#	2947#	2955#	2957#	2972#

CZRLGBO RL11/RLV11 CTLR TST 1
CZRLGB.MAC 07-DEC-79 07:39

MACY11 30A(1052) 17-DEC-79 14:01 PAGE 4-2
CROSS REFERENCE TABLE -- MACRO NAMES

MSGETT	1149#	1185#	1219#	1257#	1295#	1330#	1362#	1393#	1425#	1752#	1845#	1848#	1858#	2051#	2054#
	2063#	2066#	2098#	2101#	2393#	2457#	2503#	2551#	2596#	2599#	2698#	2955#			
MSGNGB	15#	17#	21#	22#	24#	74#	298#	393#	395#	403#	410#	418#	427#	436#	443#
	454#	506#	508#	520#	522#	532#	534#	542#	549#	551#	641#	672#	674#	699#	701#
	709#	711#	725#	974#	2926#	2928#	2951#	2953#	2973#						
MSGNIN	17#	21#	22#	401#	408#	413#	416#	422#	425#	431#	434#	441#	450#	452#	457#
	461#	463#	466#	467#	468#	469#	472#	473#	508#	522#	534#	553#	554#	555#	558#
	559#	560#	561#	569#	570#	581#	582#	614#	619#	622#	623#	630#	632#	633#	636#
	643#	652#	657#	659#	665#	668#	670#	676#	683#	693#	705#	715#	728#	729#	736#
	738#	739#	821#	978#	985#	989#	999#	1005#	1029#	1032#	1037#	1038#	1039#	1055#	1058#
	1063#	1064#	1065#	1080#	1083#	1088#	1089#	1090#	1105#	1108#	1113#	1114#	1115#	1134#	1148#
	1149#	1156#	1157#	1173#	1184#	1185#	1193#	1194#	1209#	1218#	1219#	1226#	1227#	1242#	1256#
	1257#	1264#	1265#	1280#	1294#	1295#	1300#	1301#	1317#	1329#	1330#	1335#	1336#	1351#	1361#
	1362#	1367#	1368#	1383#	1392#	1393#	1398#	1399#	1414#	1424#	1425#	1430#	1431#	1450#	1457#
	1465#	1467#	1486#	1490#	1493#	1509#	1513#	1516#	1545#	1546#	1554#	1558#	1588#	1589#	1598#
	1600#	1632#	1633#	1641#	1645#	1676#	1677#	1685#	1686#	1693#	1698#	1721#	1726#	1749#	1752#
	1758#	1760#	1767#	1769#	1776#	1780#	1804#	1810#	1812#	1817#	1836#	1839#	1845#	1848#	1856#
	1858#	1864#	1869#	1889#	1893#	1903#	1908#	1911#	1912#	1915#	1920#	1923#	1924#	1926#	1950#
	1953#	1956#	1970#	1976#	1979#	1980#	1985#	1988#	1991#	2004#	2007#	2019#	2024#	2027#	2028#
	2032#	2051#	2054#	2058#	2063#	2066#	2073#	2075#	2079#	2080#	2098#	2101#	2119#	2122#	2189#
	2190#	2196#	2214#	2217#	2220#	2240#	2243#	2246#	2250#	2268#	2276#	2277#	2281#	2287#	2289#
	2293#	2294#	2296#	2319#	2324#	2327#	2342#	2345#	2348#	2351#	2356#	2359#	2367#	2368#	2375#
	2378#	2381#	2384#	2390#	2393#	2402#	2403#	2418#	2419#	2440#	2443#	2446#	2449#	2454#	2457#
	2464#	2467#	2486#	2489#	2492#	2495#	2500#	2503#	2511#	2514#	2533#	2536#	2539#	2542#	2548#
	2551#	2558#	2561#	2581#	2584#	2587#	2590#	2596#	2599#	2606#	2609#	2631#	2634#	2672#	2676#
	2680#	2683#	2686#	2690#	2695#	2698#	2705#	2708#	2723#	2728#	2731#	2746#	2749#	2752#	2755#
	2761#	2764#	2772#	2773#	2780#	2783#	2786#	2790#	2795#	2799#	2808#	2809#	2827#	2828#	2842#
	2843#	2860#	2863#	2878#	2881#	2884#	2887#	2893#	2896#	2904#	2905#	2912#	2916#	2921#	2924#
	2928#	2930#	2931#	2932#	2933#	2934#	2935#	2937#	2953#	2954#	2955#	2956#	2957#	2973#	
MSGNLS	1156#	1193#	1226#	1264#	1300#	1335#	1367#	1398#	1430#	1864#	2079#	2418#	2842#		
MSGNTA	401#	408#	416#	425#	434#	441#	452#	461#	516#	528#	636#	670#	693#	705#	715#
	978#	1039#	1065#	1090#	1115#	1157#	1194#	1227#	1265#	1301#	1336#	1368#	1399#	1431#	1467#
	1493#	1516#	1558#	1600#	1645#	1698#	1726#	1780#	1817#	1869#	1893#	1926#	1956#	1991#	2007#
	2032#	2080#	2122#	2196#	2220#	2250#	2296#	2419#	2467#	2514#	2561#	2609#	2708#	2843#	2924#
	2937#	2957#													
MSGNTE	1019#	1044#	1070#	1095#	1120#	1162#	1199#	1232#	1270#	1306#	1341#	1373#	1404#	1436#	1472#
	1498#	1521#	1563#	1605#	1649#	1702#	1731#	1785#	1822#	1873#	1898#	1931#	1961#	1995#	2011#
	2037#	2085#	2127#	2201#	2225#	2255#	2301#	2424#	2472#	2519#	2566#	2614#	2713#	2848#	
MSHAPT	17#														
MSHNAP	17#														
MSINCR	15#	24#	74#	298#	393#	395#	401#	403#	408#	410#	413#	416#	418#	422#	425#
	427#	431#	434#	436#	441#	443#	450#	452#	454#	457#	461#	463#	466#	467#	468#
	469#	472#	473#	506#	508#	520#	522#	532#	542#	549#	551#	553#	554#	558#	560#
	569#	581#	619#	622#	623#	630#	632#	636#	641#	643#	652#	657#	659#	665#	668#
	670#	672#	674#	676#	683#	693#	699#	701#	705#	709#	711#	715#	725#	728#	736#
	738#	739#	821#	974#	989#	1005#	1019#	1029#	1032#	1037#	1038#	1039#	1044#	1055#	1058#
	1063#	1064#	1065#	1070#	1080#	1083#	1088#	1089#	1090#	1095#	1105#	1108#	1113#	1114#	1115#
	1120#	1134#	1148#	1149#	1156#	1157#	1162#	1173#	1184#	1185#	1193#	1194#	1199#	1209#	1218#
	1219#	1226#	1227#	1232#	1242#	1256#	1257#	1264#	1265#	1270#	1280#	1294#	1295#	1300#	1301#
	1306#	1317#	1329#	1330#	1335#	1336#	1341#	1351#	1361#	1362#	1367#	1368#	1373#	1383#	1392#
	1393#	1398#	1399#	1404#	1414#	1424#	1425#	1430#	1431#	1436#	1450#	1457#	1465#	1467#	1472#
	1486#	1490#	1493#	1498#	1509#	1513#	1516#	1521#	1545#	1546#	1554#	1558#	1563#	1588#	1589#
	1598#	1600#	1605#	1632#	1633#	1641#	1645#	1649#	1676#	1677#	1685#	1686#	1693#	1698#	1702#
	1721#	1726#	1731#	1749#	1752#	1758#	1760#	1767#	1769#	1776#	1780#	1785#	1804#	1810#	1812#
	1817#	1822#	1836#	1839#	1845#	1848#	1856#	1858#	1864#	1869#	1873#	1889#	1893#	1898#	1903#
	1908#	1911#	1912#	1915#	1920#	1923#	1924#	1926#	1931#	1950#	1953#	1956#	1961#	1970#	1976#

	1979#	1980#	1985#	1988#	1991#	1995#	2004#	2007#	2011#	2019#	2024#	2027#	2028#	2032#	2037#
	2051#	2054#	2058#	2063#	2066#	2073#	2075#	2079#	2080#	2085#	2098#	2101#	2119#	2122#	2127#
	2189#	2190#	2196#	2201#	2217#	2220#	2225#	2240#	2243#	2246#	2250#	2255#	2268#	2276#	2277#
	2281#	2287#	2289#	2293#	2294#	2296#	2301#	2319#	2324#	2327#	2342#	2345#	2348#	2351#	2356#
	2359#	2367#	2368#	2375#	2378#	2381#	2384#	2390#	2393#	2402#	2403#	2418#	2419#	2424#	2440#
	2443#	2446#	2449#	2454#	2457#	2464#	2467#	2472#	2486#	2489#	2492#	2495#	2500#	2503#	2511#
	2514#	2519#	2533#	2536#	2539#	2542#	2548#	2551#	2558#	2561#	2566#	2581#	2584#	2587#	2590#
	2596#	2599#	2606#	2609#	2614#	2631#	2634#	2672#	2676#	2680#	2683#	2686#	2690#	2695#	2698#
	2705#	2708#	2713#	2723#	2728#	2731#	2746#	2749#	2752#	2755#	2761#	2764#	2772#	2773#	2780#
	2783#	2786#	2790#	2795#	2799#	2808#	2809#	2827#	2828#	2842#	2843#	2848#	2860#	2863#	2878#
	2881#	2884#	2887#	2893#	2896#	2904#	2905#	2912#	2916#	2921#	2924#	2926#	2928#	2951#	2953#
M\$LDRO	554#	558#	560#	569#	581#	622#	652#	659#	668#	676#	683#	738#	1032#	1058#	1083#
	1108#	1450#	1804#	1839#	1903#	1908#	1915#	1920#	1970#	1976#	2019#	2024#	2268#	2289#	
M\$MCHI	7#														
M\$MCLO	7#														
M\$POP	19#	72#	295#	388#	401#	408#	416#	425#	434#	441#	452#	461#	504#	516#	518#
	528#	530#	536#	546#	636#	638#	670#	693#	695#	705#	707#	715#	717#	978#	1013#
	1039#	1065#	1090#	1115#	1156#	1157#	1193#	1194#	1226#	1227#	1264#	1265#	1300#	1301#	1335#
	1336#	1367#	1368#	1398#	1399#	1430#	1431#	1467#	1493#	1516#	1558#	1600#	1645#	1698#	1726#
	1780#	1817#	1864#	1869#	1893#	1926#	1956#	1991#	2007#	2032#	2079#	2080#	2122#	2196#	2220#
	2250#	2296#	2418#	2419#	2467#	2514#	2561#	2609#	2708#	2842#	2843#	2924#	2937#	2947#	2957#
	2972#														
M\$PRIN	413#	422#	431#	450#	457#	463#	466#	467#	468#	469#	472#	473#	619#	657#	665#
	736#														
M\$PUSH	15#	24#	74#	298#	393#	395#	403#	410#	418#	427#	436#	443#	454#	506#	508#
	520#	522#	532#	542#	549#	551#	641#	672#	674#	699#	701#	709#	711#	725#	974#
	1019#	1044#	1070#	1095#	1120#	1134#	1162#	1173#	1199#	1209#	1232#	1242#	1270#	1280#	1306#
	1317#	1341#	1351#	1373#	1383#	1404#	1414#	1436#	1472#	1498#	1521#	1563#	1605#	1649#	1702#
	1731#	1785#	1822#	1836#	1873#	1898#	1931#	1961#	1995#	2011#	2037#	2058#	2085#	2127#	2201#
	2225#	2255#	2301#	2319#	2424#	2472#	2519#	2566#	2614#	2713#	2723#	2848#	2926#	2928#	2951#
	2953#														
M\$PUT	413#	422#	431#	450#	457#	463#	466#	467#	468#	469#	472#	473#	619#	630#	643#
	657#	665#	736#	1029#	1055#	1080#	1105#								
M\$PUT1	413#	422#	431#	450#	457#	463#	466#	467#	468#	469#	472#	473#	619#	630#	643#
	657#	665#	736#	1029#	1055#	1080#	1105#								
M\$RADI	2930#	2931#	2932#	2933#	2934#	2935#	2954#	2956#							
M\$RNRO	581#														
M\$SETS	15#	24#	74#	298#	393#	395#	403#	410#	418#	427#	436#	443#	454#	506#	508#
	520#	522#	532#	542#	549#	551#	641#	672#	674#	699#	701#	709#	711#	725#	974#
	1019#	1044#	1070#	1095#	1120#	1134#	1162#	1173#	1199#	1209#	1232#	1242#	1270#	1280#	1306#
	1317#	1341#	1351#	1373#	1383#	1404#	1414#	1436#	1472#	1498#	1521#	1563#	1605#	1649#	1702#
	1731#	1785#	1822#	1836#	1873#	1898#	1931#	1961#	1995#	2011#	2037#	2058#	2085#	2127#	2201#
	2225#	2255#	2301#	2319#	2424#	2472#	2519#	2566#	2614#	2713#	2723#	2848#	2926#	2928#	2951#
	2953#														
M\$SVC	401#	408#	413#	416#	422#	425#	431#	434#	441#	450#	452#	457#	461#	463#	466#
	467#	468#	469#	472#	473#	553#	554#	558#	560#	569#	581#	619#	622#	623#	630#
	632#	636#	643#	652#	657#	659#	665#	668#	670#	676#	683#	693#	705#	715#	728#
	736#	738#	739#	821	989	1005	1029#	1032#	1037	1038#	1039#	1055#	1058#	1063	1064#
	1065#	1080#	1083#	1088	1089#	1090#	1105#	1108#	1113	1114#	1115#	1134#	1148	1149#	1156#
	1157#	1173#	1184	1185#	1193#	1194#	1209#	1218	1219#	1226#	1227#	1242#	1256	1257#	1264#
	1265#	1280#	1294	1295#	1300#	1301#	1317#	1329	1330#	1335#	1336#	1351#	1361	1362#	1367#
	1368#	1383#	1392	1393#	1398#	1399#	1414#	1424	1425#	1430#	1431#	1450#	1457#	1465	1467#
	1486#	1490	1493#	1509#	1513	1516#	1545	1546#	1554	1558#	1588	1589#	1598	1600#	1632
	1633#	1641	1645#	1676	1677#	1685	1686#	1693	1698#	1721#	1726#	1749#	1752#	1758	1760#
	1767	1769#	1776	1780#	1804#	1810	1812#	1817#	1836#	1839#	1845#	1848#	1856	1858#	1864#
	1869#	1889#	1893#	1903#	1908#	1911	1912#	1915#	1920#	1923	1924#	1926#	1950#	1953	1956#

	1970#	1976#	1979	1980#	1985#	1988	1991#	2004#	2007#	2019#	2024#	2027	2028#	2032#	2051#
	2054#	2058#	2063#	2066#	2073	2075#	2079#	2080#	2098#	2101#	2119	2122#	2189	2190#	2196#
	2217#	2220#	2240#	2243#	2246#	2250#	2268#	2276	2277#	2281#	2287#	2289#	2293	2294#	2296#
	2319#	2324#	2327#	2342#	2345#	2348#	2351#	2356#	2359#	2367	2368#	2375#	2378#	2381#	2384#
	2390#	2393#	2402	2403#	2418#	2419#	2440#	2443#	2446#	2449#	2454#	2457#	2464	2467#	2486#
	2489#	2492#	2495#	2500#	2503#	2511	2514#	2533#	2536#	2539#	2542#	2548#	2551#	2558	2561#
	2581#	2584#	2587#	2590#	2596#	2599#	2606	2609#	2631#	2634#	2672#	2676#	2680#	2683#	2686#
	2690#	2695#	2698#	2705	2708#	2723#	2728#	2731#	2746#	2749#	2752#	2755#	2761#	2764#	2772
	2773#	2780#	2783#	2786#	2790#	2795#	2799#	2808	2809#	2827	2828#	2842#	2843#	2860#	2863#
	2878#	2881#	2884#	2887#	2893#	2896#	2904	2905#	2912#	2916#	2921#	2924#			
MSLAB	401#	408#	413#	416#	422#	425#	431#	434#	441#	450#	452#	457#	461#	463#	466#
	467#	468#	469#	472#	473#	553#	554#	558#	560#	569#	581#	619#	622#	623#	630#
	632#	636#	643#	652#	657#	659#	665#	668#	670#	676#	683#	693#	705#	715#	728#
	736#	738#	739#	821#	989#	1005#	1029#	1032#	1037#	1038#	1039#	1055#	1058#	1063#	1064#
	1065#	1080#	1083#	1088#	1089#	1090#	1105#	1108#	1113#	1114#	1115#	1134#	1148#	1149#	1156#
	1157#	1173#	1184#	1185#	1193#	1194#	1209#	1218#	1219#	1226#	1227#	1242#	1256#	1257#	1264#
	1265#	1280#	1294#	1295#	1300#	1301#	1317#	1329#	1330#	1335#	1336#	1351#	1361#	1362#	1367#
	1368#	1383#	1392#	1393#	1398#	1399#	1414#	1424#	1425#	1430#	1431#	1450#	1457#	1465#	1467#
	1486#	1490#	1493#	1509#	1513#	1516#	1545#	1546#	1554#	1558#	1588#	1589#	1598#	1600#	1632#
	1633#	1641#	1645#	1676#	1677#	1685#	1686#	1693#	1698#	1721#	1726#	1749#	1752#	1758#	1760#
	1767#	1769#	1776#	1780#	1804#	1810#	1812#	1817#	1836#	1839#	1845#	1848#	1856#	1858#	1864#
	1869#	1889#	1893#	1903#	1908#	1911#	1912#	1915#	1920#	1923#	1924#	1926#	1950#	1953#	1956#
	1970#	1976#	1979#	1980#	1985#	1988#	1991#	2004#	2007#	2019#	2024#	2027#	2028#	2032#	2051#
	2054#	2058#	2063#	2066#	2073#	2075#	2079#	2080#	2098#	2101#	2119#	2122#	2189#	2190#	2196#
	2217#	2220#	2240#	2243#	2246#	2250#	2268#	2276#	2277#	2281#	2287#	2289#	2293#	2294#	2296#
	2319#	2324#	2327#	2342#	2345#	2348#	2351#	2356#	2359#	2367#	2368#	2375#	2378#	2381#	2384#
	2390#	2393#	2402#	2403#	2418#	2419#	2440#	2443#	2446#	2449#	2454#	2457#	2464#	2467#	2486#
	2489#	2492#	2495#	2500#	2503#	2511#	2514#	2533#	2536#	2539#	2542#	2548#	2551#	2558#	2561#
	2581#	2584#	2587#	2590#	2596#	2599#	2606#	2609#	2631#	2634#	2672#	2676#	2680#	2683#	2686#
	2690#	2695#	2698#	2705#	2708#	2723#	2728#	2731#	2746#	2749#	2752#	2755#	2761#	2764#	2772#
	2773#	2780#	2783#	2786#	2790#	2795#	2799#	2808#	2809#	2827#	2828#	2842#	2843#	2860#	2863#
	2878#	2881#	2884#	2887#	2893#	2896#	2904#	2905#	2912#	2916#	2921#	2924#			
MSSTL	401#	408#	413#	416#	422#	425#	431#	434#	441#	450#	452#	457#	461#	463#	466#
	467#	468#	469#	472#	473#	553#	554#	558#	560#	569#	581#	619#	622#	623#	630#
	632#	636#	643#	652#	657#	659#	665#	668#	670#	676#	683#	693#	705#	715#	728#
	736#	738#	739#	821#	989#	1005#	1029#	1032#	1037#	1038#	1039#	1055#	1058#	1063#	1064#
	1065#	1080#	1083#	1088#	1089#	1090#	1105#	1108#	1113#	1114#	1115#	1134#	1148#	1149#	1156#
	1157#	1173#	1184#	1185#	1193#	1194#	1209#	1218#	1219#	1226#	1227#	1242#	1256#	1257#	1264#
	1265#	1280#	1294#	1295#	1300#	1301#	1317#	1329#	1330#	1335#	1336#	1351#	1361#	1362#	1367#
	1368#	1383#	1392#	1393#	1398#	1399#	1414#	1424#	1425#	1430#	1431#	1450#	1457#	1465#	1467#
	1486#	1490#	1493#	1509#	1513#	1516#	1545#	1546#	1554#	1558#	1588#	1589#	1598#	1600#	1632#
	1633#	1641#	1645#	1676#	1677#	1685#	1686#	1693#	1698#	1721#	1726#	1749#	1752#	1758#	1760#
	1767#	1769#	1776#	1780#	1804#	1810#	1812#	1817#	1836#	1839#	1845#	1848#	1856#	1858#	1864#
	1869#	1889#	1893#	1903#	1908#	1911#	1912#	1915#	1920#	1923#	1924#	1926#	1950#	1953#	1956#
	1970#	1976#	1979#	1980#	1985#	1988#	1991#	2004#	2007#	2019#	2024#	2027#	2028#	2032#	2051#
	2054#	2058#	2063#	2066#	2073#	2075#	2079#	2080#	2098#	2101#	2119#	2122#	2189#	2190#	2196#
	2217#	2220#	2240#	2243#	2246#	2250#	2268#	2276#	2277#	2281#	2287#	2289#	2293#	2294#	2296#
	2319#	2324#	2327#	2342#	2345#	2348#	2351#	2356#	2359#	2367#	2368#	2375#	2378#	2381#	2384#
	2390#	2393#	2402#	2403#	2418#	2419#	2440#	2443#	2446#	2449#	2454#	2457#	2464#	2467#	2486#
	2489#	2492#	2495#	2500#	2503#	2511#	2514#	2533#	2536#	2539#	2542#	2548#	2551#	2558#	2561#
	2581#	2584#	2587#	2590#	2596#	2599#	2606#	2609#	2631#	2634#	2672#	2676#	2680#	2683#	2686#
	2690#	2695#	2698#	2705#	2708#	2723#	2728#	2731#	2746#	2749#	2752#	2755#	2761#	2764#	2772#
	2773#	2780#	2783#	2786#	2790#	2795#	2799#	2808#	2809#	2827#	2828#	2842#	2843#	2860#	2863#
	2878#	2881#	2884#	2887#	2893#	2896#	2904#	2905#	2912#	2916#	2921#	2924#			
MSWORD	17#	534#	821#	989#	1005#	1037#	1063#	1088#	1113#	1148#	1184#	1218#	1256#	1294#	1329#
	1361#	1392#	1424#	1465#	1490#	1513#	1545#	1554#	1588#	1598#	1632#	1641#	1676#	1685#	1693#

	1758#	1767#	1776#	1810#	1856#	1911#	1923#	1953#	1979#	1988#	2027#	2073#	2119#	2189#	2276#
	2293#	2367#	2402#	2464#	2511#	2558#	2606#	2705#	2772#	2808#	2827#	2904#	2930#	2931#	2932#
	2933#	2934#	2935#	2954#	2955#	2956#	2973								
MSXFER	2955#														
POINTE	13														
PRINTB	413	422	431	450	457	463	466	467	468	469	472	473	619	657	665
PRINTF	736														
READBU	632														
READEF	554	558	560	569											
SETPRI	676	1450	1804	1839	1903	1908	1915	1920	1970	1976	2019	2024	2268	2289	
SETVEC	630	643	1029	1055	1080	1105									
STARS	1020	1025	1047	1052	1071	1076	1096	1101	1124	1129	1164	1169	1201	1205	1233
	1238	1272	1276	1308	1313	1343	1347	1375	1379	1406	1410	1438	1447	1474	1478
	1500	1504	1523	1529	1564	1570	1608	1614	1652	1657	1706	1711	1733	1736	1787
	1797	1824	1828	1876	1882	1933	1939	1962	1967	1996	1999	2013	2016	2040	2044
	2087	2091	2130	2136	2202	2206	2228	2232	2258	2262	2306	2315	2428	2431	2475
	2478	2521	2525	2568	2572	2617	2625	2714	2719	2850	2853				
SVC	5#	7													
XFERF	2955														

. ABS. 024612 000

ERRORS DETECTED: 0

,CZRLGB.LST/CRF-SVC33/ML,CZRLGB.MAC
 RUN-TIME: 97 94 12 SECONDS
 RUN-TIME RATIO: 518/204=2.5
 CORE USED: 15K (29 PAGES)