

RLV11/RL01

DISKLESS
CVRLAA0

AH-B108A-MC

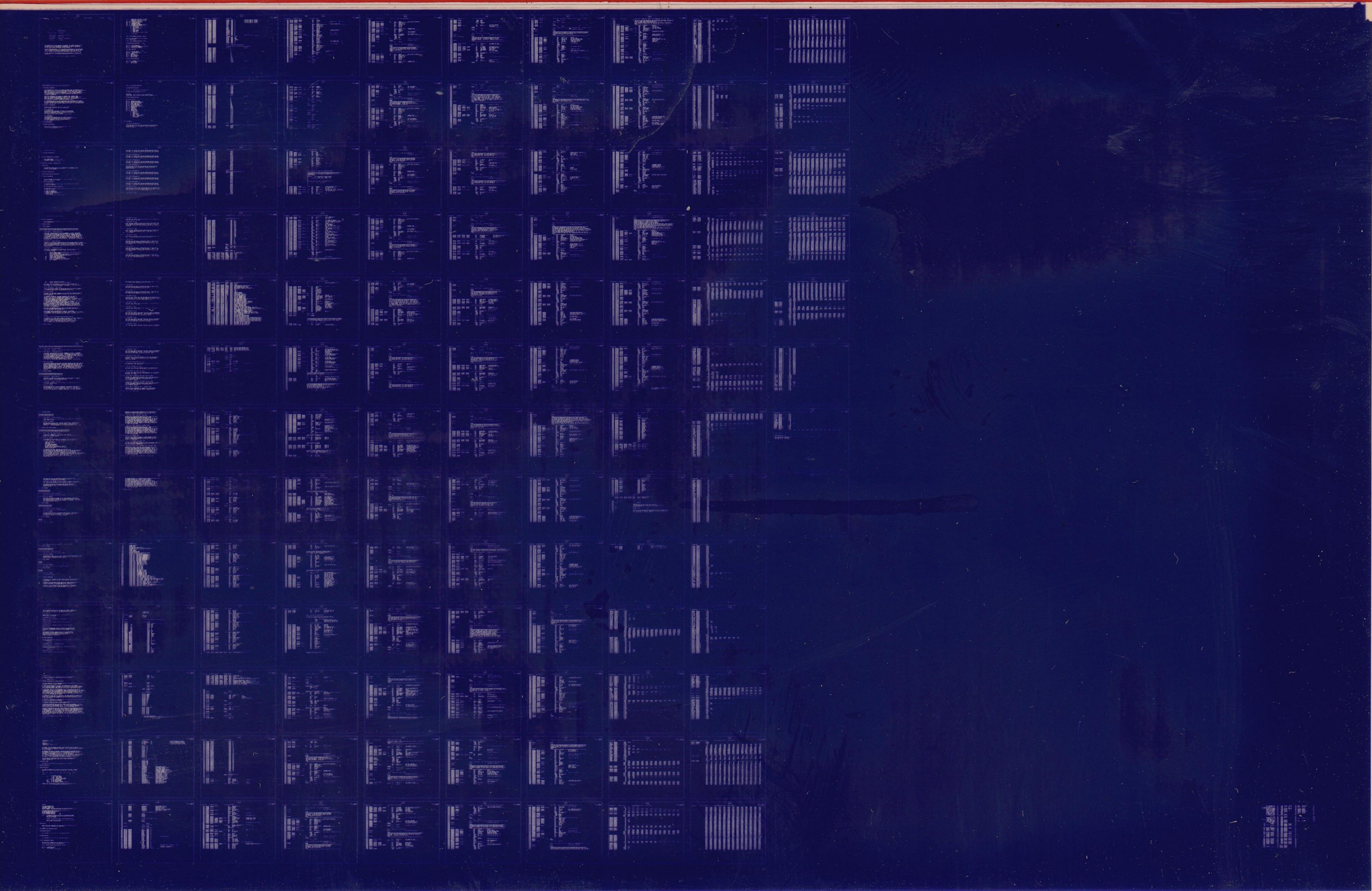
COPYRIGHT © 1978

FICHE 1 OF 1

JUN 1978

digital

MADE IN USA



EOF1CVDVCBSEQ
PDP10 PAGE: 0001

00010000

780519

PDP10 411

TOHDR1CVRLAASEQ

00010000

780519

SEQ 0001

IDENTIFICATION

PRODUCT CODE: AC-B107A-MC
PRODUCT NAME: CVRLAAD RLV11 RLO1 DSKLS
DATE CREATED: APRIL 1978
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: W. HEAVEY

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) 1978, DIGITAL EQUIPMENT CORPORATION

1.1 PROGRAM ABSTRACT FOR THE RLV11 RLO1 DISKLESS TEST

1.1.1 STRUCTURE OF PROGRAM

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

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

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

1.1.2 DIAGNOSTIC INFORMATION

THE RLV11 RLO1 DISKLESS TEST IS A LSI-11(PDP-11) BASED PROGRAM THAT WILL TEST THE CONTROLLER.

RLV11 CONTROLLER

THE PROGRAM TESTS THE BASIC INTERFACE LOGIC REGISTER MANIPULATION AND FUNCTIONALITY. THE RLV11 MAINTENANCE FUNCTION IS PERFORMED TO TEST THE CONTROLLER WRITE/READ DATA PATHS WITHOUT A DRIVE PRESENT. THIS TEST WILL RUN WITH OR WITHOUT A DRIVE PRESENT.

RL11 CONTROLLER

THE MAINTENANCE FUNCTION DOES NOT EXIST ON THE RL11 CONTROLLER. THIS PROGRAM WILL ONLY TEST THE BASIC INTERFACE LOGIC REGISTER MANIPULATION AND FUNCTIONALITY. THE "NOP" COMMAND IS THE ONLY FUNCTION PERFORMED IN THIS DIAGNOSTIC FOR THE RL11.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

PDP-11/LSI-11 PROCESSOR WITH 16K OR MORE OF CORE
CONSOLE DEVICE (LA30, LA36, VT50, ETC.)
RL11/RLV11 CONTROLLER(S) (1-8)

LINEPRINTER(OPTIONAL)

1.2.2 SOFTWARE REQUIREMENTS
-----1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
-----1.5 ASSUMPTIONS

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

2.0 OPERATING INSTRUCTIONS
-----2.1 LOADING AND STARTING PROCEDURES
-----2.1.1 LOADING PROCEDURES

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

2.1.2 STARTING PROCEDURES

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

2.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION

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

- A) LOAD THE DIAGNOSTIC
- B) START AT ADDRESS 200
- C) ANSWER THE HARDWARE QUESTIONS
- D) RECEIVE PROMPT (DS-B)
- E) ENTER STA<CR>
- F) ANSWER HARDWARE AND SOFTWARE QUESTIONS

H) TO END EXECUTION, ENTER CONTROL/C

2.2 SPECIAL ENVIRONMENTS

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

2.3 PROGRAM OPTIONS

2.3.1 START COMMAND

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

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

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

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

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

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

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

HOE	HALT ON ERROR, CAUSING COMMAND MODE TO BE ENTERED WHEN AN ERROR IS ENCOUNTERED
LOE	LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAINING THE ERROR
IER	INHIBIT ERROR REPORTING
IBE	INHIBIT BASIC ERROR REPORTS
IXE	INHIBIT EXTENDED ERROR REPORTS
PRI	DIRECT ALL MESSAGES TO A LINE PRINTER
PNT	PRINT NUMBER OF TEST BEING EXECUTED
BOE	BELL ON ERROR

ISR INHIBIT STATISTICAL REPORTS
IDR INHIBIT DROPPING OF UNITS BY DIAGNOSTIC

THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED. SEE EXAMPLE AT END OF 2.3.1.

2.3.1.4 END OF PASS SWITCH (/EOP:<INCR>)

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

2.3.1.5 EFFECT OF COMMAND

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

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

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

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

EXAMPLE:

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

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

2.3.2 RESTART COMMAND

RES(TART)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:<FLAG-LIST>/UNITS:<UNIT-LIST>

2.3.2.1 TESTS, PASS, AND FLAGS SWITCHES

<TEST-LIST>, <PASS-CNT>, AND <FLAG-LIST> ARE AS IN THE START COMMAND.

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

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

2.3.2.3 EFFECT OF COMMAND

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

2.3.3 CONTINUE COMMAND

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

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

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

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

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

2.3.3.3 EFFECT OF COMMAND

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

2.3.4 PROCEED COMMAND

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

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

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

2.3.4.2 EFFECT OF COMMAND

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

2.3.5 CREATE CORE IMAGE COMMAND

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

2.3.5.1 TESTS, PASS, AND FLAGS SWITCHES

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

2.3.5.2 EFFECT OF COMMAND

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

```
INVOKE THE XXDP UTILITY UPD1 OR UPD2
LOAD XXN:FILE.BIN
START 200
<QUESTIONS AND ANSWERS>
GET PTAB END MESSAGE
LET MACHINE COME TO HALT
RESTART UTILITY USING RESTART ADDRESS
CHANGE HICORE ADDRESS TO PTAB END VALUE
DUMP XXN:FILE.BIC
```

THE OPERATOR DIALOGUE (HARDWARE AND SOFTWARE) WILL BE EXECUTED AS IN THE START COMMAND, BUT AT THE END OF THE QUESTIONS THE HALT STATE WILL BE ENTERED, AT WHICH TIME THE OPERATOR SHOULD DUMP THE PROGRAM TO THE XXDP LIBRARY USING A BIC EXTENSION TO INDICATE THAT THIS FILE IS CHAINABLE. HE SHOULD USE THE XXDP UTILITY "UPD1" OR "UPD2" TO DO THIS. IF THE P-TABLES EXTEND BEYOND 14.5K, A MESSAGE WILL BE ISSUED GIVING THE NEW UPPER CORE LIMIT, TO WHICH THE OPERATOR MUST ADJUST BEFORE DUMPING. HE MAY NOW DELETE THE NON-CHAINABLE BIN FILE IF DESIRED, SINCE THE BIC FILE HAS ALL THE CAPABILITIES OF IT.

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

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

NOTE THAT ALTHOUGH THE CHAINABLE IMAGE CAN BE EXECUTED ON A 16K MACHINE, THE ORIGINAL CCI CREATION MUST BE DONE ON A LARGER MACHINE, THE EXACT SIZE BEING DEPENDENT ON WHICH UPDATE UTILITY IS USED.

2.3.6 ADD COMMAND

ADD/UNITS:<UNIT-LIST>

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

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

2.3.6.2 EFFECT OF COMMAND

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

2.3.7 DROP COMMAND

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

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

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

2.3.7.2 EFFECT OF COMMAND

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

2.3.8 PRINT COMMAND

PRI(NT)

2.3.8.1 EFFECT OF COMMAND

ISR (INHIBIT STATISTICAL REPORTING) FLAG IS CLEARED.

2.3.9 DISPLAY COMMAND

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

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

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

2.3.9.2 EFFECT OF COMMAND

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

2.3.10 FLAGS COMMAND

FLA(GS)

2.3.10.1 EFFECT OF COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

2.3.11 ZFLAGS COMMAND

ZFL(AGS)

2.3.11.1 EFFECT OF COMMAND

ALL FLAGS ARE CLEARED.

2.3.12 CONTROL CHARACTERS

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

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

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

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 (0) 174400?

ANSWER WITH THE BUS ADDRESS OF THE CONTROLLER.

VECTOR (0) 330?

ANSWER WITH THE INTERRUPT VECTOR OF THE CONTROLLER.

BR LEVEL (0) 5?

ANSWER WITH THE INTERRUPT PRIORITY OF THE CONTROLLER.

DRIVE (0) 0?

SINCE THIS PROGRAM RUNS WITHOUT A DRIVE, THIS QUESTION DOES
NOT APPLY. THE HARDWARE QUESTION ON DRIVE NUMBER IS ASKED
TO MAINTAIN COMPATIBILITY WITH THE RL11 PROGRAMS FOR CHAIN
MODE.

WHEN TESTING MULTIPLE CONTROLLERS(0 TO 7), THE OPERATOR
CAN RESPOND TO DRIVE NUMBER BY TYPING A NUMBER(0-7) FOR
EACH CONTROLLER. THEN WHEN AN ERROR IS PRINTED, THE DRIVE
NUMBER IN THE ERROR PRINTOUT WILL REFER TO THE
NUMBER ASSIGNED THE CONTROLLER.

2.3.14 SOFTWARE PARAMETERS

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

CHANGE SW?

IS ANSWERED YES(Y). THE QUESTIONS ARE:

DROP ON ERROR LIMIT (L) Y?

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

ANSWER Y OR N

ERROR LIMIT (D) 10?

NUMBER OF ERRORS ALLOWED BEFORE DROPPING UNIT.

AUTOSIZE (L) N?

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

ANSWER Y OR N

2.3.15 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

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

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

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

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

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

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

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

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

THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:

UNITS (D) ? 64

UNIT 1

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

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

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

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

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

2.4 EXECUTION TIMES

ONE PASS OF THE PROGRAM TAKES APPROXIMATELY 45 SECONDS.

3.0 ERROR INFORMATION

3.1 ERROR REPORTING

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

CVRL? 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

REGISTER DESCRIPTIONS CAN BE FOUND IN SECTION 5.0.

CS: CONTROL AND STATUS REGISTER
BA: BUS ADDRESS REGISTER
DA: DISK ADDRESS REGISTER
MP: MULTIPURPOSE REGISTER

NOTE: TO PREVENT EXTENSIVE PRINTOUTS ON BUFFER FAILURES
USE THE "FLAG:IXE" (INHIBIT EXTENDED ERROR REPORTS)
SUPERVISOR COMMAND.

EXAMPLE: DS-B>STA/FLAG:IXE OR DS-B>RES/FLAG:IXE

USE OF THIS FLAG WILL PRINT ONLY THE FIRST FAILURE
ENCOUNTERED IN THE BUFFER.

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 (XXXXXD)

BIT 15 - COMPOSITE ERROR
BIT 14 - DRIVE ERROR
BIT 13 - NON EXISTENT MEMORY ERROR

BIT 11 - DATA LATE (WITH BIT 10 CLEAR)
 BIT 11 - HEADER CRC (WITH BIT 10 SET)
 BIT 10 - 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 (POP-11) MAINT (LSI-11)
 1 - WRITE CHECK
 2 - GET DRIVE STATUS
 3 - SEEK
 4 - READ HEADER
 5 - WRITE DATA
 6 - READ DATA
 7 - READ WITHOUT HEADER COMPARE

BIT 0 - DRIVE READY

RLBA - BUS ADDRESS REGISTER (XXXXX2)

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

RLDA - DISK ADDRESS REGISTER (XXXXX4)

FOR READ/WRITE FUNCTIONS

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

FOR SEEK FUNCTION

BIT 15 - MUST BE ZERO(0)
 BIT 14-7 - DIFFERENCE TO NEW CYLINDER
 BIT 6-5 - MUST BE ZERO(0)
 BIT 4 - SURFACE
 BIT 3 - MUST BE ZERO
 BIT 2 - SEEK DIRECTION(1 - IN / 0 - OUT)
 BIT 1 - MUST BE ZERO
 BIT 0 - MUST BE ONE(1)

FOR GET STATUS FUNCTION

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

RLMP - MULTIPURPOSE REGISTER

FOR READ/WRITE FUNCTION

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

FOR READ AFTER MAINTENANCE FUNCTION

BIT 15-0

FIRST RLMP: CRC OF STARTING DISK ADDRESS VALUE+3

SECOND RLMP: CRC OF CRC OF STARTING DISK ADDRESS VALUE+4

FOR GET STATUS FUNCTION

HAS DRIVE STATUS

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

5.0 TEST SUMMARIES

TEST 1 - RLCS WRITE 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.

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

TEST 3 - RLDA WRITE 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 WRITE 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 - RLCS READ 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 6 - RLBA READ ADDRESSABILITY

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

TEST 7 - RLDA READ 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 8 - RLMP READ 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 9 - 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) WHICH WILL BE SET IF BIT 14 (DRIVE ERROR) IS SET.

TEST 10 - BUS RESET OF RLBA

THIS TEST WILL VERIFY THAT THE BUS RESET OF THE PROCESSOR WILL

TEST 11 - BUS RESET OF RLDA

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

TEST 12 - 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 13 - 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 14 - 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 15 - 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 0 ARE DON'T CARES.

TEST 16 - 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 0 ARE DON'T CARES.

TEST 17 - 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 18 - BIC OF RLBA

READ-MODIFY-WRITE SEQUENCE OF THE RLBA WORKS. BITS 15-0 ARE BIT CLEARED USING GROWING AND WALKING 0'S AND 1'S.

TEST 19 - 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 20 - 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 21 - 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) WHICH WILL BE SET IF BIT 14 (DRIVE ERROR) IS SET.

TEST 22 - BUS RESET OF RLBA

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

TEST 23 - BUS RESET OF RLDA

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

TEST 24 - 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 25 - 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 26 - 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

AND RLBA ARE VERIFIED THAT THEY DID NOT CHANGE.

TEST 27 - 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 28 - NOOP FUNCTION (RL11 ONLY)

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 29 - TEST NOOP DOES NOTHING (RL11 ONLY)

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

TEST 30 - TEST OF INTERRUPT (RL11 ONLY)

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

TEST 31 - TEST PRIORITY BR LEVEL (RL11 ONLY)

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 32 - RLV11 MAINT. FORCED OPI TEST, LESS THAN 510 WORDS

PERFORM RLV11 MAINTENANCE FUNCTION 0 WITH LESS THAN 510 WORDS TO FORCE OPI ERROR. THE TEST SHOULD FORCE COMPOSITE ERROR (BIT 15), HEADER NOT FOUND (BIT 12) AND OPI (BIT 10). DRIVE ERROR IS IGNORED.

TEST 33 - RLV11 MAINT. FORCED OPI TEST, MORE THAN 511 WORDS

PERFORM RLV11 MAINTENANCE FUNCTION 0 WITH MORE THAN 511 WORDS TO FORCE OPI ERROR. THE TEST SHOULD FORCE COMPOSITE ERROR (BIT 15), HEADER NOT FOUND (BIT 12) AND OPI (BIT 10). DRIVE ERROR IS IGNORED.

TEST 34 - RLV11 MAINT. FORCED OPI TEST - INTERRUPT MODE

PERFORM TEST OF INTERRUPT BY ISSUING RLV11 MAINTENANCE FUNCTION 0 WITH MORE THAN 511 WORDS TO FORCE OPI ERROR. CHECK INTERRUPT OPERATION AND REPORT IF ERROR FOUND.

TEST 35 - RLV11 OPI TIMEOUT TEST

PERFORM RLV11 MAINTENANCE FUNCTION 0 WITH INTERRUPT MODE.
FORCE OPI TIMEOUT BY SETTING WORD COUNT TO -512 WORDS.
MEASURE OPI TIMEOUT AND COMPARE TO MIN. AND MAX. LIMITS.

TEST 36 - TEST RLV11 MAINT. FUNCTION -FLAG MODE

PERFORM RLV11 MAINTENANCE FUNCTION 0 (FLAG MODE) AND CHECK
DA AND BA REGISTERS FOR PROPER INCREMENT. CHECK
THE SERIAL WRITE/READ DATA PATHS BY READING OUT OF THE FIFO
VIA THE MP REGISTER THE CRC OF THE DA+3 AND THE CRC OF THE
CRC OF THE DA+4 AND COMPARING TO EXPECTED RESULTS. CHECK
THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE FIFO
INTO BUF2 MEMORY FOR PROPER DATA. CHECK THE PREVIOUSLY
WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR A VALUE=123456
TO INSURE THAT THE DATA TRANSFER IN MAINTENANCE MODE WAS
NOT MORE THAN 255 WORDS.

TEST 37 - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

PERFORM RLV11 MAINTENANCE FUNCTION 0 (INT. MODE) AND CHECK
DA AND BA REGISTERS FOR PROPER INCREMENT. CHECK
THE SERIAL WRITE/READ DATA PATHS BY READING OUT OF THE FIFO
VIA THE MP REGISTER THE CRC OF THE DA+3 AND THE CRC OF THE
CRC OF THE DA+4 AND COMPARING TO EXPECTED RESULTS. CHECK
THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE FIFO
INTO BUF2 MEMORY FOR PROPER DATA. CHECK THE PREVIOUSLY
WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR A VALUE=123456
TO INSURE THAT THE DATA TRANSFER IN MAINTENANCE MODE WAS
NOT MORE THAN 255 WORDS.

TEST 38 - RLV11 FIFO ADDRESS TEST

TEST THAT FIFO OPERATES CORRECTLY. STORE ADDRESS PATTERN
IN BUF1 (0-255) WHICH CONTAINS A UNIQUE PATTERN IN EACH
LOCATION. PERFORM MAINTENANCE FUNCTION AND TEST BUF2 FOR
PROPER FIFO ADDRESSING.

TEST 39 - RLV11 FIFO ADDRESS COMPLEMENT TEST

TEST THAT FIFO ADDRESSES CORRECTLY. STORE THE ADDRESS COMPLEMENT
OF 0-255 INTO BUF1. PERFORM MAINTENANCE FUNCTION AND CHECK
BUF2 FOR PROPER FIFO ADDRESSING.

TEST 40 - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INTERRUPT MODE

PERFORM RLV11 MAINT. FUNCTION WITH COMPLEMENT DATA IN BUF1.
CHECK DA AND BA REGISTERS FOR PROPER INCREMENT. CHECK
THE SERIAL WRITE/READ DATA PATHS BY READING OUT OF THE FIFO
VIA THE MP REGISTER THE CRC OF THE DA+3 AND THE CRC OF THE
CRC OF THE DA+4 AND COMPARING TO EXPECTED RESULTS. CHECK
THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE FIFO
INTO BUF2 MEMORY FOR PROPER DATA. CHECK THE PREVIOUSLY
WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR A VALUE=123456
TO INSURE THAT THE DATA TRANSFER IN MAINTENANCE MODE WAS
NOT MORE THAN 255 WORDS.

TEST 41 - TEST RLV11 MAINT. WITH RANDOM DATA -INTERRUPT MODE

THE RANDOM PATTERN IS THE SAME FOR EACH CONTROLLER UNDER TEST.
THE RANDOM PATTERN WILL CHANGE AT END OF PASS.
THE RANDOM PATTERN WILL INIT AT START OR RESTART.
CHECK DA AND BA REGISTERS FOR PROPER INCREMENT. CHECK
THE SERIAL WRITE/READ DATA PATHS BY READING OUT OF THE FIFO
VIA THE MP REGISTER THE CRC OF THE DA+3 AND THE CRC OF THE
CRC OF THE DA+4 AND COMPARING TO EXPECTED RESULTS. CHECK
THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE FIFO
INTO BUF2 MEMORY FOR PROPER DATA. CHECK THE PREVIOUSLY
WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR A VALUE=123456
TO INSURE THAT THE DATA TRANSFER IN MAINTENANCE MODE WAS
NOT MORE THAN 255 WORDS.

7.0 PROGRAM LISTING

CVRLAA.P11

14-APR-78 15:04

TABLE OF CONTENTS

74	GLOBAL EQUATES
193	GLOBAL DATA
197	GLOBAL DATA
257	PATTERNS FOR REGISTER R/W
332	PATTERNS FOR MAINT. CRC TEST OF SERIAL DATA PATH
431	BUFFERS FOR RLV11 MAINTENANCE FUNCTION
437	GLOBAL TEXT
441	GLOBAL ERRORS
766	INITIALIZATION CODE
941	GLOBAL SUBROUTINES
972	ROUTINE TO CHECK FOR CONTROLLER ERRORS
1345	ROUTINE TO CALCULATE CRC
1467	**TEST 1** - RLCS WRITE ADDRESSABILITY
1509	**TEST 2** - RLBA WRITE ADDRESSABILITY
1552	**TEST 3** - RLDA WRITE ADDRESSABILITY
1594	**TEST 4** - RLMP WRITE ADDRESSABILITY
1635	**TEST 5** - RLCS READ ADDRESSABILITY
1677	**TEST 6** - RLBA READ ADDRESSABILITY
1720	**TEST 7** - RLDA READ ADDRESSABILITY
1762	**TEST 8** - RLMP READ ADDRESSABILITY
1803	**TEST 9** - BUS RESET OF RLCS
1850	**TEST 10** - BUS RESET OF RLBA
1885	**TEST 11** - BUS RESET OF RLDA
1917	**TEST 12** - READ WRITE OF RLCS
1972	**TEST 13** - READ WRITE OF RLBA
2021	**TEST 14** - READ WRITE OF RLDA
2067	**TEST 15** - BIS OF RLCS
2118	**TEST 16** - BIC OF RLCS
2167	**TEST 17** - BIS OF RLBA
2215	**TEST 18** - BIC OF RLBA
2260	**TEST 19** - BIS OF RLDA
2304	**TEST 20** - BIC OF RLDA
2349	**TEST 21** - BUS RESET OF RLCS
2396	**TEST 22** - BUS RESET OF RLBA
2431	**TEST 23** - BUS RESET OF RLDA
2463	**TEST 24** - UNIQUENESS OF RLCS
2518	**TEST 25** - UNIQUENESS OF RLBA
2573	**TEST 26** - UNIQUENESS OF RLDA
2630	**TEST 27** - UNIQUENESS OF RLMP
2701	**TEST 28** - NOOP FUNCTION(RL11 ONLY)
2735	**TEST 29** - TEST NOOP DOES NOTHING (RL11 ONLY)
2810	**TEST 30** - TEST OF INTERRUPT (RL11 ONLY)
2858	**TEST 31** - TEST PRIORITY BR LEVEL (RL11 ONLY)
2929	**TEST 32** - RLV11 MAINT. FORCED OPI TEST, LESS THAN 510 WORDS
2977	**TEST 33** - RLV11 MAINT. FORCED OPI TEST, MORE THAN 511 WORDS
3025	**TEST 34** - RLV11 MAINT. FORCED OPI TEST - INTERRUPT MODE
3083	**TEST 35** - RLV11 OPI TIMEOUT TEST
3190	**TEST 36** - TEST RLV11 MAINT. FUNCTION - FLAG MODE
3350	**TEST 37** - TEST RLV11 MAINT. FUNCTION - INTERRUPT MODE
3526	**TEST 38** - RLV11 FIFO ADDRESS TEST
3630	**TEST 39** - RLV11 FIFO ADDRESS COMPLEMENT TEST
3734	**TEST 40** - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE
3909	**TEST 41** - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE
4174	DIAGNOSTIC SUPERVISOR -- LOW CORE SET UP

CVRLAA.P11 14-APR-78 15:04

1			.ENABLE	AMA
2			.ENABLE	ABS
3			.NLIST	ME,CND,MD
4			SVC	
5			SVCINS=0	
6			SVCTAG=0	
7			.=2000	
8	000000	000000		
9		000000		
10		002000		
11				
12				
13	002000		POINTER	BGNSFT,BGNSW,BGNDU,BGNAU
14				
15				
16	002000		BGNMOD	MDHEDR
17				
18	002000		HEADER	CVRLA,A,0,,.20.,RL01
19	002000	103	.ASCII	2C3
20	002001	126	.ASCII	2V3
21	002002	122	.ASCII	2R3
22	002003	114	.ASCII	2L3
23	002004	101	.ASCII	2A3
24	002005	000	.BYTE	00
25	002006	000	.BYTE	00
26	002007	000	.BYTE	00
27	002010	101	.ASCII	2A3
28	002011	060	.ASCII	203
29	002012	000000	.WORD	00
30	002014	000024	.WORD	20.
31	002016	027552	.WORD	L\$HARD
32	002020	027676	.WORD	L\$SOFT
33	002022	013314	.WORD	L\$HW
34	002024	013330	.WORD	L\$SW
35	002026	030114	.WORD	L\$LAST
36	002030	000000	.WORD	00
37	002032	000000	.WORD	00
38	002034	000000	.WORD	00
39	002036	000000	.WORD	00
40	002040	013340	.WORD	L\$DISPATCH
41	002042	000000	.WORD	00
42	002044	000000	.WORD	00
43	002046	000000	.WORD	00
44	002050	002	.BYTE	C\$REVISION
45	002051	000	.BYTE	C\$EDIT
46	002052	000000	.WORD	
47	002054	000000	.WORD	
48	002056	000000	.WORD	0
49	002060	000000	.WORD	00
50	002062	000000	.WORD	00
51	002064	002114	.WORD	L\$DVTYP
52	002066	000000	.WORD	00
53	002070	002112	.WORD	L\$DR
54	002072	002112	.WORD	L\$DRST
55	002074	014212	.WORD	L\$AU
56	002076	014206	.WORD	L\$DU

CVRLAA.P11 14-APR-78 15:04

```

57 002100 000014      .WORD 14
58 002102 000000      .WORD 0
59 002104 013462      .WORD L$INIT
60 002106 014144      .WORD L$CLEAN
61
62 002110      ENDMOD
63
64
65
66 002110      DEVREG
67 002110 000000      .WORD 0
68 002112 000001      .BLKW
69
70 002114      DEVTYP <RL01>
71 002114 046122 030460 000 .ASCIZ &RL01a
72      002122 .EVEN
73
74      .SBTTL GLOBAL EQUATES
75 002122 BGNMOD GLBEQAT
76
77 002122      EQUALS
78
79      ; BIT DIFINITIONS
80
81      100000      BIT15== 100000
82      040000      BIT14== 40000
83      020000      BIT13== 20000
84      010000      BIT12== 10000
85      004000      BIT11== 4000
86      002000      BIT10== 2000
87      001000      BIT09== 1000
88      000400      BIT08== 400
89      000200      BIT07== 200
90      000100      BIT06== 100
91      000040      BIT05== 40
92      000020      BIT04== 20
93      000010      BIT03== 10
94      000004      BIT02== 4
95      000002      BIT01== 2
96      000001      BIT00== 1
97
98      001000      BIT9== BIT09
99      000400      BIT8== BIT08
100     000200      BIT7== BIT07
101     000100      BIT6== BIT06
102     000040      BIT5== BIT05
103     000020      BIT4== BIT04
104     000010      BIT3== BIT03
105     000004      BIT2== BIT02
106     000002      BIT1== BIT01
107     000001      BIT0== BIT00
108
109      ; EVENT FLAG DEFINITIONS
110      EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
111      EF16:EF01 AVAILABLE FOR PROGRAM USE
112

```


CVRLAA.P11

14-APR-78 15:04

GLOBAL EQUATES

113	000040	EF.START==	32.	:	START COMMAND WAS ISSUED
114	000037	EF.RESTART==	31.	:	RESTART COMMAND WAS ISSUED
115	000036	EF.CONTINUE==	30.	:	CONTINUE COMMAND WAS ISSUED
116	000035	EF.NEW==	29.	:	A NEW PASS HAS BEEN STARTED
117	000034	EF.PWR==	28.	:	A POWER-FAIL/POWER-UP OCCURRED
118					
119	000020	EF16==	16.		
120	000017	EF15==	15.		
121	000016	EF14==	14.		
122	000015	EF13==	13.		
123	000014	EF12==	12.		
124	000013	EF11==	11.		
125	000012	EF10==	10.		
126	000011	EF09==	9.		
127	000010	EF08==	8.		
128	000007	EF07==	7.		
129	000006	EF06==	6.		
130	000005	EF05==	5.		
131	000004	EF04==	4.		
132	000003	EF03==	3.		
133	000002	EF02==	2.		
134	000001	EF01==	1.		
135					
136		:			
137		:			
138	000340	PRI07==	340		
139	000300	PRI06==	300		
140	000240	PRI05==	240		
141	000200	PRI04==	200		
142	000140	PRI03==	140		
143	000100	PRI02==	100		
144	000040	PRI01==	40		
145	000000	PRI00==	0		
146	000001	DRDY=BIT0		:	DRIVE READY (RLCS)
147	000100	INTEN=BIT6		:	INTERRUPT ENABLE (RLCS)
148	100000	ERR=BIT15		:	RL11 ERROR (RLCS)
149	040000	DERR=BIT14		:	RL01 DRIVE ERROR (RLCS)
150	002000	OPI=BIT10		:	OPERATION INCOMPLETE (RLCS)
151	000200	CRDY=BIT7		:	CONTROLLER READY (RLCS)
152	000040	BA17=BIT5		:	EXTENDED ADDRESS BIT 17 (RLCS)
153	000020	BA16=BIT4		:	EXTENDED ADDRESS BIT 16 (RLCS)
154	020000	NXM=BIT13		:	NON-EXISTANT MEMORY (RLCS)
155	000000	DS0=0		:	DRIVE SELECT 0 (RLCS)
156	000400	DS1=BIT8		:	DRIVE SELECT 1 (RLCS)
157	001000	DS2=BIT9		:	DRIVE SELECT 2 (RLCS)
158	001400	DS3=BIT8!BIT9		:	DRIVE SELECT 3 (RLCS)
159	000000	NOOP0=0		:	FUNCTION-NOOP(0)-RL11
160	000016	NOOP7=BIT1!BIT2!BIT3		:	FUNCTION-NOOP(7)-RL11
161	000000	MAINT=0		:	MAINTENANCE FUNCTION-RLV11
162	000002	WRCHK=BIT1		:	WRITE CHECK FUNCTION
163	000004	GSTAT=BIT2		:	GET STATUS FUNCTION
164	000006	SEEK=BIT2!BIT1		:	SEEK FUNCTION
165	000010	RDHDR=BIT3		:	READ HEADER FUNCTION
166	000012	WRITE=BIT3!BIT1		:	WRITE DATA FUNCTION
167	000014	READ=BIT3!BIT2		:	READ DATA FUNCTION
168	000202	GODRV=BIT1!BIT7		:	CRDY AND DRDY

CVRLAA.P11 14-APR-78 15:04

GLOBAL EQUATES

```

169      000010
170      000002
171      000001
172      000004
173      000100
174      000100
175      000020
176
177
178
179      000000
180      000002
181      000004
182      000006
183      000010
184
185
186
187      000000
188      000002
189      000004
190
191      002122
192
193
194
195      002122
196
197
198
199      002122 000000
200      002124 000000
201      002126 000000
202      002130 000000
203      002132 000000
204      002134 000000
205      002136 000000
206      002140 000000
207      002142 000000
208      002144 000000
209      002146 000000
210      002150 000000
211      002152 000000
212      002154 000000
213      002156 000000
214      002160 000000
215      002162 000000
216      002164 000000
217      002166 000000
218      002170 000000
219      002172 000000
220      002174 000000
221      002176 000000
222      002200 000000
223      002202 120001
224      002204 000004

```

```

DRST=BIT3      ;DRIVE RESET (RLDA)
GSBIT=BIT1     ;GET STATUS BIT (RLDA)
MK=BIT0        ;MARKER BIT (RLDA)
SIGN=BIT2      ;SIGN BIT (RLDA)
RHHS=BIT6      ;HEAD SELECT IN READ HEADER
STHS=BIT6      ;HEAD SELECT IN STATUS BACK
DAHS=BIT4      ;HEAD SELECT IN SEEK

;OFFSET FOR HARDWARE P-TABLE

CSR=0
VECT=2
PRIOR=4
DRBT=6
CNT=10

;OFFSET FOR SOFTWARE P-TABLE

DLT=0
ELT=2
SIZE=4

ENDMOD

.SBTTL GLOBAL DATA

BGNMOD GLBDAT

.SBTTL GLOBAL DATA

UUT: .WORD 0
UNITST: .WORD 0
RLCS: .WORD 0
RLBA: .WORD 0
RLDA: .WORD 0
RLMP: .WORD 0
BCSR: .WORD 0
BPRIOR: .WORD 0
BVEC: .WORD 0
DRIVE: .WORD 0
B.CS: .WORD 0
B.BA: .WORD 0
B.DA: .WORD 0
B.MP: .WORD 0
DERFLG: .WORD 0
E.CS: .WORD 0
E.BA: .WORD 0
E.DA: .WORD 0
E.MP: .WORD 0
E.MP1: .WORD 0
PFLG: .WORD 0
TRPFLG: .WORD 0
INTFLG: .WORD 0
LDCSR: .WORD 0
XPOLY: .WORD 120001
ERRVEC: .WORD 4

```

;DRIVE UNDER TEST

;PROCESSOR TYPE 0=UNIBUS 1=Q-BUS

;INTERRUPT OCCURANCE FLAG
;LOCATION TO FORM RLCS

CVRLAA.P11 14-APR-78 15:04

GLOBAL DATA

```

225 002206 000000
226 002210 000000
227 002212 000000
228 002214 000000
229 002216 000000
230 002218 000000
231 002220 000000
232 002222 000000
233 002224 000000
234 002226 000000
235 002228 000000
236 002230 000000
237 002232 000000
238 002234 000000
239 002236 000000
240 002238 000000
241 002240 000000
242 002242 000000
243 002244 000000
244 002246 000000
245 002248 000000
246 002250 000000
247 002252 000000
248 002254 000000
249 002256 000000
250 002258 000000
251 002260 000000
252 002262 000000
253 002264 000000
254 002266 000000
255 002268 000000
256 002270 000000
257 002272 000000
258 002274 000000
259 002276 000000
260 002278 000000
261 002280 000000
262 002282 000000
263 002284 000000
264 002286 000000
265 002288 000000
266 002290 000000
267 002292 000000
268 002294 000000
269 002296 000000
270 002298 000000
271 002300 000000
272 002302 000000
273 002304 000000
274 002306 000001
275 002310 000003
276 002312 000007
277 002314 000017
278 002316 000037
279 002320 000077
280 002322 000177
281 002324 000377
282 002326 000777
283 002330 001777
284 002332 003777
285 002334 007777
286 002336 017777
287 002340 037777
288 002342 077777
289 002344 177777
290 002346 177776
291 002350 177774
292 002352 177770

```

```

BCCFBK: .WORD 0
CALBCC: .WORD 0
TEMP2: .WORD 0
TEMP3: .WORD 0
TEMP4: .WORD 0
TEMP5: .WORD 0
TEMP1: .WORD 0
TMP0: .WORD 0
TMP1: .WORD 0
TMP2: .WORD 0
CHECK: .WORD 0
GDDAT: .WORD 0
BDDAT: .WORD 0
GCRCPT: .WORD 0
GDCRCA: .WORD 0
GDCRCB: .WORD 0
GDDATP: .WORD 0
GDATMP: .WORD 0
MATFLG: .WORD 0
ERRLMT: .WORD 0
WHY: .WORD 0
T.CNTRL: .WORD 0
TMPFNC: .WORD 0
OPIMN: .WORD 155
OPIMX: .WORD 650
HINUM: .WORD 176543
LONUM: .WORD 123456
TEMLO: .WORD 0
TEMHI: .WORD 0
PATSAV: .WORD 0
SAVCNT: .WORD 0

```

```

;LOCATION USED BY "SIMBCC"
;LOCATION USED BY "SIMBCC"
;LOCATION USED BY "SIMBCC"
;LOCATION USED BY "SIMBCC"
;LOCATION USED BY "SIMBCC"

```

;REASON FOR DROP IN AUTOSIZE

```

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

```

BEGPAT: 0 ;GROWING 1

```

1
3
7
17
37
77
177
377
777
1777
3777
7777
17777
37777
77777
177777
177776
177774
177770

```

;GROWING 0

CVRLAA.P11 14-APR-78 15:04

PATTERNS FOR REGISTER R/W

281	002354	177760	177760
282	002356	177740	177740
283	002360	177700	177700
284	002362	177600	177600
285	002364	177400	177400
286	002366	177000	177000
287	002370	176000	176000
288	002372	174000	174000
289	002374	170000	170000
290	002376	160000	160000
291	002400	140000	140000
292	002402	100000	100000
293			
294	002404	000000	000000
295	002406	000001	1
296	002410	000002	2
297	002412	000004	4
298	002414	000010	10
299	002416	000020	20
300	002420	000040	40
301	002422	000100	100
302	002424	000200	200
303	002426	000400	400
304	002430	001000	1000
305	002432	002000	2000
306	002434	004000	4000
307	002436	010000	10000
308	002440	020000	20000
309	002442	040000	40000
310	002444	100000	100000
311	002446	177777	177777
312	002450	177776	177776
313	002452	177775	177775
314	002454	177773	177773
315	002456	177767	177767
316	002460	177757	177757
317	002462	177737	177737
318	002464	177677	177677
319	002466	177577	177577
320	002470	177377	177377
321	002472	176777	176777
322	002474	175777	175777
323	002476	173777	173777
324	002500	167777	167777
325	002502	157777	157777
326	002504	137777	137777
327	002506	077777	077777
328	002510	177777	177777
329	002512	000000	000000
330			
331			
332			
333	002514	155552	155552
334	002516	133330	133330
335	002520	066663	066663
336	002522	125247	125247

;WALKING 1

;WALKING 0

ENDPAT: 000000

SBTTL PATCRC: PATTERNS FOR MAINT. CRC TEST OF SERIAL DATA PATH

155552
133330
066663
125247

D03

PAGE: 0029AS
SEQ 0029

CVRLAA.P11

14-APR-78 15:04

PATTERNS FOR MAINT. CRC TEST OF SERIAL DATA PATH

337	002524	052522	052522
338	002526	177774	177774
339	002530	000374	000374
340	002532	022217	022217
341	002534	044441	044441
342	002536	166663	166663
343	002540	144441	144441
344	002542	033330	033330
345	002544	011106	011106
346	002546	070704	070704
347	002550	107065	107065
348	002552	111106	111106
349	002554	167353	167353
350	002556	156732	156732
351	002560	146311	146311
352	002562	135670	135670
353	002564	114626	114626
354	002566	104205	104205
355	002570	073564	073564
356	002572	063143	063143
357	002574	042101	042101
358	002576	031460	031460
359	002600	021037	021037
360	002602	010416	010416
361	002604	000000	000000

CRCEND: 000000

:DATA PATTERNS FOR MAINTENANCE TEST
PATDAT: 155555

364	002606	155555	155555
365	002610	133333	133333
366	002612	066666	066666
367	002614	125252	125252
368	002616	052525	052525
369	002620	177777	177777
370	002622	000000	000000
371	002624	107070	107070
372	002626	070707	070707
373	002630	144444	144444
374	002632	033333	033333
375	002634	011111	011111
376	002636	022222	022222
377	002640	044444	044444
378	002642	111111	111111
379	002644	166666	166666
380	002646	010421	010421
381	002650	021042	021042
382	002652	031463	031463
383	002654	042104	042104
384	002656	063146	063146
385	002660	073567	073567
386	002662	104210	104210
387	002664	114631	114631
388	002666	135673	135673
389	002670	146314	146314
390	002672	156735	156735
391	002674	167356	167356
392	002676	000000	000000

ENDDAT: 000000

CVRLAA.P11

14-APR-78 15:04

PATTERNS FOR MAINT. CRC TEST OF SERIAL DATA PATH

393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439

002700 000000
002702 000002
002704 000004
002706 000010
002710 000020
002712 000040
002714 000100
002716 000400
002720 001000
002722 001576
002724 001574
002726 001570
002730 001560
002732 001540
002734 001500
002736 001400
002740 001576
002742 001574
002744 001566
002746 001556
002750 001536
002752 001436
002754 001136
002756 000076
002760 000006
002762 000016
002764 000036
002766 000076
002770 000176
002772 000576
002774 001576
002776 000000
003000 000240

003500 000400
004500 000400
005500

005500 047516 041440 047117
005516 047516 042040 044522
005541 040 051104 000126
005546 047040 046530 000
005553 040 050117 000111
005560 044040 051103 000103
005566 044040 043116 000
005573 040 041504 000113
005600 042040 052114 000
005605 105 050130 042047

;PATTERNS FOR TEST OF RLCS

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

.SBTTL BUFFERS FOR RLVI1 MAINTENANCE FUNCTION
 BUF1: .BLKW 256.
 BUF2: .BLKW 256.
 ENDMOD

BGNMOD GLBTXT
 .SBTTL GLOBAL TEXT

NORES: .ASCIZ /NO CONTROLLER/
 NODRY: .ASCIZ /NO DRIVE CONNECTED/
 DEMES: .ASCIZ / DRV/
 NXMES: .ASCIZ / NXM/
 OPIMES: .ASCIZ / OPI/
 HCRCHES: .ASCIZ / HCRC/
 HNFMES: .ASCIZ / HNF/
 DCKMES: .ASCIZ / DCK/
 DLTMES: .ASCIZ / DLT/
 EXPMES: .ASCIZ /EXP'D: COMP HNF OPI REC'D: /

CVRLAA.P11

14-APR-78 15:04

GLOBAL TEXT

005642	047516	042440	050130	NONMES:	.ASCIZ	/NO EXPECTED ERRORS FOUND/
005673	015	000012		MSCRLF:	.ASCIZ	<15><12>
005676	000015			LF:	.ASCIZ	<15>
005700	041440	046517	000120	COMP:	.ASCIZ	/COMP/
005706	047506	041522	042105	OPIERR:	.ASCIZ	/FORCED OPI(GET STATUS) CAUSED OTHER ERRORS/
005761	116	047517	020120	NOPMES:	.ASCIZ	/NOOP OPERATION-FLAG MODE/
006012	047516	050117	047440	NOPINT:	.ASCIZ	/NOOP OPERATION-INTR. MODE/
006044	040515	047111	042524	MATMES:	.ASCIZ	/MAINTENANCE OPERATION-FLAG MODE/
006104	040515	047111	042524	MATINT:	.ASCIZ	/MAINTENANCE OPERATION-INTERRUPT MODE/
006151	103	035123	000040	ARLCS:	.ASCIZ	/CS: /
006156	041040	035101	000040	ARLBA:	.ASCIZ	/BA: /
006164	042040	035101	000040	ARLDA:	.ASCIZ	/DA: /
006172	046440	035120	000040	ARLMP:	.ASCIZ	/MP: /
006200	042502	047506	042522	BEREG:	.ASCIZ	/BEFORE COMMAND: /
006221	124	046511	020105	AFREG:	.ASCIZ	/TIME OF ERROR: /
006242	047503	052116	047522	CRTIM:	.ASCIZ	/CONTROLLER TIMED OUT/
006267	104	044522	042526	DRTIM:	.ASCIZ	/DRIVE READY TIMED OUT/
006315	103	047101	047040	EM1:	.ASCIZ	/CAN NOT ADDRESS RLCS/
006342	040503	020116	047516	EM2:	.ASCIZ	/CAN NOT ADDRESS RLBA/
006367	103	047101	047040	EM3:	.ASCIZ	/CAN NOT ADDRESS RLDA/
006414	040503	020116	047516	EM4:	.ASCIZ	/CAN NOT ADDRESS RLMP/
006441	122	041514	020123	EM5:	.ASCIZ	/RLCS READ/WRITE ERROR (BIT 0 DON'T CARE)%
006512	046122	040502	051040	EM6:	.ASCIZ	/RLBA READ/WRITE ERROR%
006540	046122	040504	051040	EM7:	.ASCIZ	/RLDA READ/WRITE ERROR%
006566	046122	040502	042440	EM10:	.ASCIZ	/RLBA ERROR AFTER MAINT. FUNCTION/
006627	117	044520	053440	EM11:	.ASCIZ	/OPI WOULD NOT GENERATE INTERRUPT/
006670	046122	040504	042440	EM12:	.ASCIZ	/RLDA ERROR AFTER MAINT. FUNCTION/
006731	116	020117	047111	EM13:	.ASCIZ	/NO INTERRUPT FROM NOOP(0)/
006763	116	047517	024120	EM14:	.ASCIZ	/NOOP(0) MODIFIED RLMP/
007011	116	047517	024120	EM15:	.ASCIZ	/NOOP(0) MODIFIED RLBA/
007037	116	047517	024120	EM16:	.ASCIZ	/NOOP(0) MODIFIED RLDA/
007065	111	052116	051105	EM17:	.ASCIZ	/INTERRUPT PRIORITY FAILURE/
007120	046122	050115	020072	EM20:	.ASCIZ	/RLMP: CRC OF DA+3 ERROR (SERIAL DATA PATH)/
007173	122	046514	035120	EM21:	.ASCIZ	/RLMP: CRC OF CRC OF DA+4 ERROR (SERIAL DATA PATH)/
007255	115	044501	052116	EM22:	.ASCIZ	/XMAINT. FILL/EMPTY FIFO DMA DATA TRANSFER COMPARE ERROR%
007344	040515	047111	042524	EM23:	.ASCIZ	/MAINTENANCE LAST WORD+1 FAILURE/
007404	047516	044440	052116	EM24:	.ASCIZ	/NO INTERRUPT FROM MAINT. FUNCTION/
007446	040515	047111	042524	EM25:	.ASCIZ	/MAINTENANCE FIFO ADDRESS ERROR/
007505	115	044501	052116	EM26:	.ASCIZ	/MAINTENANCE FIFO ADDRESS COMPLEMENT ERROR/
007557	115	044501	052116	EM27:	.ASCIZ	/MAINT. FORCED OPI ERROR, LESS THAN 510 WORDS/
007633	115	044501	052116	EM30:	.ASCIZ	/MAINT. FORCED OPI ERROR, MORE THAN 511 WORDS/
007707	117	044520	052040	EM31:	.ASCIZ	/OPI TIMING ERROR/
007730	051127	052111	047111	EM44:	.ASCIZ	/WRITING RLMP MODIFIED RLCS/
007763	127	044522	044524	EM45:	.ASCIZ	/WRITING RLMP MODIFIED RLBA/
010016	051127	052111	047111	EM46:	.ASCIZ	/WRITING RLMP MODIFIED RLDA/
010051	102	052111	051440	EM61:	.ASCIZ	/BIT SET INSTRUCTION ON RLCS YIELDED WRONG RESULT/
010132	044502	020124	046103	EM62:	.ASCIZ	/BIT CLEAR INSTRUCTION ON RLCS YIELDED WRONG RESULT/
010215	102	052111	051440	EM63:	.ASCIZ	/BIT SET INSTRUCTION ON RLBA YIELDED WRONG RESULT/
010276	044502	020124	046103	EM64:	.ASCIZ	/BIT CLEAR INSTRUCTION ON RLBA YIELDED WRONG RESULT/
010361	102	052111	051440	EM65:	.ASCIZ	/BIT SET INSTRUCTION ON RLDA YIELDED WRONG RESULT/
010442	044502	020124	046103	EM66:	.ASCIZ	/BIT CLEAR INSTRUCTION ON RLDA YIELDED WRONG RESULT/
010525	102	051525	051040	EM67:	.ASCIZ	/BUS RESET DID NOT CLEAR RLCS/
010562	052502	020123	042522	EM70:	.ASCIZ	/BUS RESET DID NOT CLEAR RLBA/
010617	102	051525	051040	EM71:	.ASCIZ	/BUS RESET DID NOT CLEAR RLDA/
010654	051127	052111	047111	EM72:	.ASCIZ	/WRITING RLCS MODIFIED RLBA/
010707	127	044522	044524	EM73:	.ASCIZ	/WRITING RLCS MODIFIED RLDA/

CVRLAA.P11 14-APR-78 15:04

GLOBAL TEXT

010742	051127	052111	047111	EM74:	.ASCIZ	/WRITING RLBA MODIFIED RLCS/
010774	051127	052111	047111	EM75:	.ASCIZ	/WRITING RLBA MODIFIED RLDA/
011026	051127	052111	047111	EM76:	.ASCIZ	/WRITING RLDA MODIFIED RLCS/
011061	127	044522	044524	EM77:	.ASCIZ	/WRITING RLDA MODIFIED RLBA/
011114	046122	051503	041440	EM101:	.ASCIZ	/RLCS CONTAINED FOLLOWING ERROR(S): /
011161	000170			EM102:	.BLKB	120.
	011352				.EVEN	
(0)	011352				ENDMOD	
440					.SBTTL	GLOBAL ERRORS
441					BGNMOD	GLBERR
442	011352				BGNMSG	ERRO
443						
444	011352					
445					JSR	PC,LINE1
446	011352	004737	012042		JSR	PC,LINE2
447	011356	004737	012076			
448					JSR	RS,CKERLT
449	011362	004537	014216			;CHECK ERROR LIMIT
450	011366				ENDMSG	
451	011366			L10000:		
452	011366	104023			EMT	C\$MSG
453						
454	011370				BGNMSG	ERR1
455						
456	011370	004737	012042		JSR	PC,LINE1
457						
458	011374	004537	014216		JSR	RS,CKERLT
459	011400					;CHECK ERROR LIMIT
460	011400				ENDMSG	
461	011400			L10001:		
462	011400	104023			EMT	C\$MSG

CVRLAA.P11 14-APR-78 15:04

GLOBAL ERRORS

```

463
464 011402
465
466 011402 004737 012042
467 011406
468 011406 013746 002236
469 011412 013746 002234
470 011416 012746 012534
471 011422 012746 000003
472 011426 010600
473 011430 104014
474 011432 062706 000010
475 011436 004537 014216
476 011442
477 011442
478 011442 104023
479
480
481 011444
482 011444 004737 012042
483 011450 004737 012076
484 011454
485 011454 012746 012655
486 011460 012746 000001
487 011464 010600
488 011466 104014
489 011470 062706 000004
490 011474
491 011474 013746 002236
492 011500 013746 002234
493 011504 013746 002224
494 011510 013746 002164
495 011514 013746 002162
496 011520 012746 013217
497 011524 012746 000006
498 011530 010600
499 011532 104014
500 011534 062706 000016
501 011540 004537 014216
502 011544
503 011544
504 011544 104023
505
506
507
508 011546
509
510 011546 004737 012042
511 011552 004737 012076
512 011556
513 011556 013746 002236
514 011562 013746 002234
515 011566 012746 012534
516 011572 012746 000003
517 011576 010600
518 011600 104014

```

BGNMSG ERR2

```

JSR PC,LINE1
PRINTB #FRMT4,GDDAT,BDDAT
MOV BDDAT,-(SP)
MOV GDDAT,-(SP)
MOV #FRMT4,-(SP)
MOV #3,-(SP)
MOV SP,RO
EMT CSPNTB
ADD #10,SP
JSR RS,CKERLT
ENDMSG

```

L10002:

EMT CSMSG

BGNMSG ERR3

```

JSR PC,LINE1
JSR PC,LINE2
PRINTB #FRMT99
MOV #FRMT99,-(SP)
MOV #1,-(SP)
MOV SP,RO
EMT CSPNTB
ADD #4,SP
PRINTB #FRMT14,E.BA,E.DA,TMPO,GDDAT,BDDAT
MOV BDDAT,-(SP)
MOV GDDAT,-(SP)
MOV TMPO,-(SP)
MOV E.DA,-(SP)
MOV E.BA,-(SP)
MOV #FRMT14,-(SP)
MOV #6,-(SP)
MOV SP,RO
EMT CSPNTB
ADD #16,SP
JSR RS,CKERLT
ENDMSG

```

L10003:

EMT CSMSG

BGNMSG ERR4

```

JSR PC,LINE1
JSR PC,LINE2
PRINTB #FRMT4,GDDAT,BDDAT
MOV BDDAT,-(SP)
MOV GDDAT,-(SP)
MOV #FRMT4,-(SP)
MOV #3,-(SP)
MOV SP,RO
EMT CSPNTB

```

CVRLAA.P11 14-APR-78 15:04

GLOBAL ERRORS

519 011602 062706 000010
 520 011606 004537 014216
 521 011612
 522 011612
 523 011612 104023
 524 011614
 525 011614 004737 012042
 526 011620 004537 014216
 527 011624
 528 011624 104023
 529 011626
 530 011626 004737 012042
 531 011632 004737 012314
 532 011636 004737 012076
 533 011642
 534 011642 012746 012655
 535 011646 012746 000001
 536 011652 010600
 537 011654 104014
 538 011656 062706 000004
 539 011662 004537 014216
 540 011666
 541 011666 104023
 542 011670
 543 011670 004737 012042
 544 011674
 545 011674 013746 002236
 546 011700 012746 012731
 547 011704 012746 000002
 548 011710 010600
 549 011712 104014
 550 011714 062706 000006
 551 011720 004537 014216
 552 011724
 553 011724
 554 011724 104023
 555 011726
 556 011726 004737 012042
 557 011732 004737 012076
 558 011736 004737 012366
 559 011742

ADD #10, SP
 JSR RS, CKERLT ;CHECK ERROR LIMIT
 ENDMSG
 L10004:
 EMT CSMSG
 BGNMSG ERR5
 JSR PC, LINE1
 JSR RS, CKERLT ;CHECK ERROR LIMIT
 ENDMSG
 L10005:
 EMT CSMSG
 BGNMSG ERR6
 JSR PC, LINE1
 JSR PC, LINE3
 JSR PC, LINE2
 1\$:
 PRINTB #FRMT99
 MOV #FRMT99, -(SP)
 MOV #1, -(SP)
 MOV SP, R0
 EMT CSPTB
 ADD #4, SP
 JSR RS, CKERLT ;CHECK ERROR LIMIT
 ENDMSG
 L10006:
 EMT CSMSG
 BGNMSG ERR7
 JSR PC, LINE1
 PRINTB #FRMT6, BDDAT
 MOV BDDAT, -(SP)
 MOV #FRMT6, -(SP)
 MOV #2, -(SP)
 MOV SP, R0
 EMT CSPTB
 ADD #6, SP
 JSR RS, CKERLT
 ENDMSG
 L10007:
 EMT CSMSG
 BGNMSG ERR10
 JSR PC, LINE1
 JSR PC, LINE2
 JSR PC, LINE4
 PRINTB #FRMT99

CVRLAA.P11 14-APR-78 15:04

GLOBAL ERRORS

```

575 011742 012746 012655
576 011746 012746 000001
577 011752 010600
578 011754 104014
579 011756 062706 000004
580 011762 004537 014216
581 011766
582 011766
583 011766 104023
584
585 011770
586 011770 004737 012042
587 011774 004737 012076
588 012000
589 012000 013746 002236
590 012004 013746 002266
591 012010 013746 002264
592 012014 012746 012766
593 012020 012746 000004
594 012024 010600
595 012026 104014
596 012030 062706 000012
597 012034 004537 014216
598 012040
599 012040
600 012040 104023
601
602
603 012042
604 012042 005046
605 012044 153716 002145
606 012050 013746 002126
607 012054 012746 012414
608 012060 012746 000003
609 012064 010600
610 012066 104014
611 012070 062706 000010
612 012074 000207
613
614 012076
615 012076 013746 002150
616 012102 012746 006156
617 012106 013746 002146
618 012112 012746 006151
619 012116 012746 006200
620 012122 012746 012454
621 012126 012746 000006
622 012132 010600
623 012134 104014
624 012136 062706 000016
625 012142
626 012142 013746 002154
627 012146 012746 006172
628 012152 013746 002152
629 012156 012746 006164
630 012162 012746 012473

```

```

MOV #FRMT99, -(SP)
MOV #1, -(SP)
MOV SP, RO
EMT C$PNTB
ADD #4, SP
JSR RS, CKERLT
ENDMSG
L10010: EMT C$MSG
BGNMSG ERR11
JSR PC, LINE1
JSR PC, LINE2
PRINTB #FRMT10, OPIMN, OPIMX, BDDAT
MOV BDDAT, -(SP)
MOV OPIMX, -(SP)
MOV OPIMN, -(SP)
MOV #FRMT10, -(SP)
MOV #4, -(SP)
MOV SP, RO
EMT C$PNTB
ADD #12, SP
JSR RS, CKERLT
ENDMSG
L10011: EMT C$MSG
LINE1: PRINTB #FRMT1, RLCS, <B, DRIVE+1>
CLR -(SP)
BISB DRIVE+1, (SP)
MOV RLCS, -(SP)
MOV #FRMT1, -(SP)
MOV #3, -(SP)
MOV SP, RO
EMT C$PNTB
ADD #10, SP
RTS PC
LINE2: PRINTB #FRMT2, #BEREG, #ARLCS, B.CS, #ARLBA, B.BA
MOV B.BA, -(SP)
MOV #ARLBA, -(SP)
MOV B.CS, -(SP)
MOV #ARLCS, -(SP)
MOV #BEREG, -(SP)
MOV #FRMT2, -(SP)
MOV #6, -(SP)
MOV SP, RO
EMT C$PNTB
ADD #16, SP
PRINTB #FRMT2A, #ARLDA, B.DA, #ARLMP, B.MP
MOV B.MP, -(SP)
MOV #ARLMP, -(SP)
MOV B.DA, -(SP)
MOV #ARLDA, -(SP)
MOV #FRMT2A, -(SP)

```

CVRLAA.P11 14-APR-78 15:04

GLOBAL ERRORS

```

631 012166 012746 000005
632 012172 010600
633 012174 104014
634 012176 062706 000014
635 012202
636 012202 013746 002162
637 012206 012746 006156
638 012212 013746 002160
639 012216 012746 006151
640 012222 012746 006221
641 012226 012746 012454
642 012232 012746 000006
643 012236 010600
644 012240 104014
645 012242 062706 000016
646 012246
647 012246 013746 002170
648 012252 013746 002166
649 012256 012746 006172
650 012262 013746 002164
651 012266 012746 006164
652 012272 012746 012506
653 012276 012746 000006
654 012302 010600
655 012304 104014
656 012306 062706 000016
657 012312 000207
658
659 012314
660 012314 012746 011114
661 012320 012746 012527
662 012324 012746 000002
663 012330 010600
664 012332 104014
665 012334 062706 000006
666 012340
667 012340 012746 011161
668 012344 012746 012527
669 012350 012746 000002
670 012354 010600
671 012356 104014
672 012360 062706 000006
673 012364 000207
674
675 012366
676 012366 012746 011161
677 012372 012746 012527
678 012376 012746 000002
679 012402 010600
680 012404 104014
681 012406 062706 000006
682 012412 000207
683
684

```

```

MOV #5, -(SP)
MOV SP, R0
EMT C$PNTB
ADD #14, SP
PRINTB #FRMT2, #AFREG, #ARLCS, E.CS, #ARLBA, E.BA
MOV E.BA, -(SP)
MOV #ARLBA, -(SP)
MOV E.CS, -(SP)
MOV #ARLCS, -(SP)
MOV #AFREG, -(SP)
MOV #FRMT2, -(SP)
MOV #6, -(SP)
MOV SP, R0
EMT C$PNTB
ADD #16, SP
PRINTB #FRMT2B, #ARLDA, E.DA, #ARLMP, E.MP, E.MP1
MOV E.MP1, -(SP)
MOV E.MP, -(SP)
MOV #ARLMP, -(SP)
MOV E.DA, -(SP)
MOV #ARLDA, -(SP)
MOV #FRMT2B, -(SP)
MOV #6, -(SP)
MOV SP, R0
EMT C$PNTB
ADD #16, SP
RTS PC

```

```

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

```

```

LINE4: PRINTB #FRMT3, #EM102
MOV #EM102, -(SP)
MOV #FRMT3, -(SP)
MOV #2, -(SP)
MOV SP, R0
EMT C$PNTB
ADD #6, SP
RTS PC

```

```

012414 040445 047503 052116 FRMT1: .ASCIZ /%ACONTROLLER: %06%A DRIVE: %01/
012454 047045 052045 052045 FRMT2: .ASCIZ /%N%T%T%06%T%06/

```


CVRLAA.P11 14-APR-78 15:04

GLOBAL ERRORS

012473	045	022524	033117	FRMT2A: .ASCIZ	/XT%06%T%06/
012506	052045	047445	022466	FRMT2B: .ASCIZ	/XT%06%T%06%A %06/
012527	045	022516	000124	FRMT3: .ASCIZ	/XN%T/
012534	047045	040445	054105	FRMT4: .ASCIZ	/XN%EXP'D: %06%A REC'D: %06%N/
012572	047045	042045	022463	FRMT9B: .ASCIZ	/XN%D3%A WORDS BAD OUT OF 255 WORDS TRANSFERRED%N%N/
012655	045	000116		FRMT99: .ASCIZ	/XN/
012660	047045	040445	040514	FRMT5: .ASCIZ	/XN%ALAST: %06%A PRES: %06%A EXP'D: %06%N/
012731	045	022516	040501	FRMT6: .ASCIZ	/XN%AAT PROCESSOR LEVEL %06%N/
012766	047045	040445	040522	FRMT10: .ASCIZ	/XN%ARANGE %D3%A - %D3%A MILLISECONDS WAS %D6%N/
013046	040445	051105	047522	FRMT11: .ASCIZ	/XAERROR LIMIT EXCEEDED-DROPPED%N/
013107	045	022516	042101	FRMT12: .ASCIZ	/XN%ADRIE DID NOT RECOVER FROM POWER FAILURE%N/
013166	047045	052045	040445	FRMT13: .ASCIZ	/XN%TXA - WILL NOT TEST%N/
013217	045	041101	035101	FRMT14: .ASCIZ	/XABA: %06%A DA: %06%A ADDR: %06%A EXP'D: %06%A REC'D %06%N/

.EVEN

685					
686					
687	013312			ENDMOD	
688					
689	013312			BGNMOD	HPTCODE
690				BGNHW	
691	013312	000005		.WORD	L10012-LSHW/2
692	013312	174400		.WORD	174400
693	013314	000330		.WORD	330
694	013316	000240		.WORD	240
695	013320	000000		.WORD	0
696	013322	000000		.WORD	0
697	013324	000001		.WORD	1
698					
699	013326			ENDHW	
700	013326			L10012:	
701					
702	013326			ENDMOD	
703					
704	013326			BGNMOD	SPTCODE
705				BGNSW	
706	013326	000003		.WORD	L10013-LSSW/2
707	013326				
708				DROP:	.WORD 0
709	013330	000000		MERLMT:	.WORD 10.
710	013332	000012		T.SIZE:	.WORD 0
711	013334	000000			
712				ENDSW	
713	013336			L10013:	
714	013336				
715				ENDMOD	
716	013336				
717				BGNMOD	DSPCODE
718	013336				
719				DISPATCH	41
720	013336	000051		.WORD	41
721	013336	016432		.WORD	T1
722	013340				

CVRLAA.P11 14-APR-78 15:04

GLOBAL ERRORS

723	013342	016530	.WORD	T2
724	013344	016626	.WORD	T3
725	013346	016724	.WORD	T4
726	013350	017022	.WORD	T5
727	013352	017116	.WORD	T6
728	013354	017212	.WORD	T7
729	013356	017306	.WORD	T8
730	013360	017402	.WORD	T9
731	013362	017512	.WORD	T10
732	013364	017564	.WORD	T11
733	013366	017622	.WORD	T12
734	013370	017742	.WORD	T13
735	013372	020044	.WORD	T14
736	013374	020132	.WORD	T15
737	013376	020256	.WORD	T16
738	013400	020402	.WORD	T17
739	013402	020506	.WORD	T18
740	013404	020606	.WORD	T19
741	013406	020676	.WORD	T20
742	013410	020776	.WORD	T21
743	013412	021106	.WORD	T22
744	013414	021160	.WORD	T23
745	013416	021216	.WORD	T24
746	013420	021342	.WORD	T25
747	013422	021502	.WORD	T26
748	013424	021642	.WORD	T27
749	013426	022046	.WORD	T28
750	013430	022076	.WORD	T29
751	013432	022302	.WORD	T30
752	013434	022366	.WORD	T31
753	013436	022532	.WORD	T32
754	013440	022642	.WORD	T33
755	013442	022752	.WORD	T34
756	013444	023106	.WORD	T35
757	013446	023436	.WORD	T36
758	013450	024250	.WORD	T37
759	013452	025126	.WORD	T38
760	013454	025474	.WORD	T39
761	013456	026046	.WORD	T40
762	013460	026724	.WORD	T41

ENDMOD

.SBTTL	INITIALIZATION CODE
BGNMOD	INITCODE

BGNINIT

BRESET

EMT CSRESET

REDEF #EF.PWR

MOV #EF.PWR, R0

EMT CSREFG

BCOMplete CONT

BCS CONT

REDEF #EF.RESTART

;POWER UP?????

;BRANCH

;RESTART?

NOPWR:

000034

CVRLAA.P11 14-APR-78 15:04

INITIALIZATION CODE

779	013474	012700	000037		MOV	#EF.RESTART,RO	
780	013500	104050			EMT	CSREFG	
781	013502				BCOMPLETE	START	
782	013502	103411			BCS	START	
783	013504				READEF	#EF.START	;START?
784	013504	012700	000040		MOV	#EF.START,RO	
785	013510	104050			EMT	CSREFG	
786	013512				BCOMPLETE	START	
787	013512	103405			BCS	START	
788	013514				READEF	#EF.NEW	;NEW PASS????
789	013514	012700	000035		MOV	#EF.NEW,RO	
790	013520	104050			EMT	CSREFG	
791	013522				BCOMPLETE	START1	;YES, THEN RE INIT
792	013522	103411			BCS	START1	
793	013524	000425			BR	CONTINUE	
794	013526	012737	176543	002270	START: MOV	#176543,HINUM	;RANDOM GEN. PRIME
795	013534	012737	123456	002272	MOV	#123456,LONUM	;RANDOM GEN. PRIME
796	013542	005037	002254		CLR	ERRLMT	;CLEAR ERROR LIMIT
797	013546	013737	002012	002122	START1: MOV	LSUNIT,UUT	
798	013554	012737	177777	002124	MOV	#-1,UNITST	
799	013562	013737	002272	002274	MOV	LONUM,TEMLO	
800	013570	013737	002270	002276	MOV	HINUM,TEMHI	;NEW PRIMES AT END OF PASS
801	013576	000404			BR	NXT	
802							
803	013600				CONTINUE: READEF	#EF.CONTINUE	;CONTINUE????
804	013600	012700	000036		MOV	#EF.CONTINUE,RO	
805	013604	104050			EMT	CSREFG	
806	013606				BCOMPLETE	CONT	
807	013606	103433			BCS	CONT	
808							
809	013610	005737	002122		NXT: TST	UUT	;DONE ALL UUT'S
810	013614	001006			BNE	IS	;NO
811	013616	012737	177777	002124	MOV	#-1,UNITST	
812	013624	013737	002012	002122	MOV	LSUNIT,UUT	
813							
814	013632	005237	002124		IS: INC	UNITST	
815	013636	005337	002122		DEC	UUT	
816	013642				REST: GPHARD	UNITST,RO	
817	013642	013700	002124		MOV	UNITST,RO	
818	013646	104042			EMT	CSGPHRD	
819	013650				BCOMPLETE	NXT	
820	013650	103357			BCC	NXT	
821	013652	012037	002136		IS: MOV	(RO)+,BCSR	
822	013656	012037	002142		MOV	(RO)+,BVEC	
823	013662	012037	002140		MOV	(RO)+,BPRIOR	
824	013666	012037	002144		MOV	(RO)+,DRIVE	
825	013672	012037	002260		MOV	(RO)+,T.CNTRL	;GET CONTROLLER TYPE
826							
827	013676	013737	002274	002272	CONT: MOV	TEMLO,LONUM	;RESTORE RANDOM FOR NEXT UUT
828	013704	013737	002276	002270	MOV	TEMHI,HINUM	;RESTORE PRIME FOR NEXT UUT
829	013712	013700	002136		MOV	BCSR,RO	
830	013716	010037	002126		MOV	RO,RLCS	
831	013722	062700	000002		ADD	#2,RO	
832	013726	010037	002130		MOV	RO,RLBA	
833	013732	062700	000002		ADD	#2,RO	
834	013736	010037	002132		MOV	RO,RLDA	

CVRLAA.P11

14-APR-78 15:04

INITIALIZATION CODE

835	013742	062700	000002		ADD	#2,RO	
836	013746	010037	002134		MOV	RO,RLMP	
837	013752	005737	013334		TST	T.SIZE	;DO WE WANT TO CHECK UNITS??
838	013756	001450			BEQ	END	;NO
839							
840	013760	005037	002174		CLR	TRPFLG	;CLR OUT TRAP FLAG
841	013764				SETVEC	ERRVEC,STRPHAN,#340	;SETUP VECTOR TO CATCH NON-EXIST
842	013764	012746	000340		MOV	#340,-(SP)	
843	013770	012746	016264		MOV	STRPHAN,-(SP)	
844	013774	013746	002204		MOV	ERRVEC,-(SP)	
845	014000	012746	000003		MOV	#3,-(SP)	
846	014004	104037			EMT	CSSVEC	
847	014006	062706	000010		ADD	#10,SP	
848	014012	005777	166110		TST	RLCS	;ACCESS CONTROLLER
849	014016				CLRVEC	ERRVEC	;RELEASE VECTOR
850	014016	013700	002204		MOV	ERRVEC,RO	
851	014022	104036			EMT	CSCVEC	
852	014024	005737	002174		TST	TRPFLG	;DID IT TRAP
853	014030	001423			BEQ	END	
854	014032	012737	005500	002256	MOV	#NORES,WHY	;SETUP ERR MESS
855	014040			BS:	PRINTB	#FRMT13,WHY	
856	014040	013746	002256		MOV	WHY,-(SP)	
857	014044	012746	013166		MOV	#FRMT13,-(SP)	
858	014050	012746	000002		MOV	#2,-(SP)	
859	014054	010600			MOV	SP,RO	
860	014056	104014			EMT	CSPNTB	
861	014060	062706	000006		ADD	#6,SP	
862	014064	004737	012042	BS:	JSR	PC,LINE1	;GIVE DRIVE INFO
863	014070				DODU	UNITST	;TELL SUPERVISOR TO DROP IT
864	014070	013700	002124		MOV	UNITST,RO	
865	014074	104053			EMT	CSDODU	
866	014076				DOCLN		;FORCE AN ABORT
867	014076	104044			EMT	CSDCLN	
868	014100			END:	SETVEC	BVEC,#INTSRV,#340	
869	014100	012746	000340		MOV	#340,-(SP)	
870	014104	012746	016272		MOV	#INTSRV,-(SP)	
871	014110	013746	002142		MOV	BVEC,-(SP)	
872	014114	012746	000003		MOV	#3,-(SP)	
873	014120	104037			EMT	CSSVEC	
874	014122	062706	000010		ADD	#10,SP	
875	014126	005037	002172		CLR	PFLG	;CLR PROCESSOR FLAG
876	014132				READBUS		;Q-BUS
877	014132	104007			EMT	CSDDBU	
878	014134				BNCOMPLETE	IS	
879	014134	103002			BCC	IS	
880	014136	005237	002172		INC	PFLG	;NO, Q-BUS THEN
881	014142			IS:			
882	014142				ENDINIT		
883	014142			L10014:			
884	014142	104011			EMT	CSINIT	
885							
886	014144				ENDMOD		
887							
888	014144			BGNMOD	CLNCODE		
889							
890	014144				BGNCLN		

CVRLAA.P11

14-APR-78 15:04

INITIALIZATION CODE

891							
892	014144					SETPRI	#PRIO0
893	014144	012700	000000			MOV	#PRIO0,RO
894	014150	104041				EMT	CSSPRI
895							
896	014152	032777	000200	165746	15:	BIT	#CRDY,ARLCS
897	014160	001774				BEQ	15
898							
899	014162					SETPRI	#PRIO7
900	014162	012700	000340			MOV	#PRIO7,RO
901	014166	104041				EMT	CSSPRI
902	014170	042777	000100	165730		BIC	#INTEN,ARLCS
903							
904	014176					CLRVEC	BVEC
905	014176	013700	002142			MOV	BVEC,RO
906	014202	104036				EMT	CSCVEC
907	014204				25:		
908	014204					ENDCLN	
909	014204				L10015:		
910	014204	104012				EMT	CSCLEAN
911							
912	014206					ENDMOD	
913							
914							
915							
916	014206				BGNMOD	DRPCODE	
917							
918	014206					BGNDU	
919							
920	014206	000240				NOP	
921							
922	014210					ENDDU	
923	014210				L10016:		
924	014210	104055				EMT	CSDU
925							
926	014212				ENDMOD		
927							
928	014212				BGNMOD	ADDCODE	
929							
930	014212					BGNAU	
931							
932	014212	000240				NOP	
933							
934	014214					ENDAU	
935	014214				L10017:		
936	014214	104054				EMT	CSAU
937							
938	014216				ENDMOD		
939							
940							
941					.SBTTL	GLOBAL	SUBROUTINES
942							
943	014216				BGNMOD	GLBSUB	
944							
945	014216				CKERLT:	INLOOP	
946	014216	104020				EMT	CSINLP

CVRLAA.P11 14-APR-78 15:04

GLOBAL SUBROUTINES

```

947 014220          BCOMPLETE      99$
948 014220 103427    BCS      99$
949 014222 005737 013330    TST      DROP
950 014226 001424          BEQ      99$
951 014230 005237 002254    INC      ERRLMT
952 014234 023737 002254 013332    CMP      ERRLMT, MERLMT
953 014242 002416          BLT      99$
954
955 014244          PRINTF    #FRMT11
956 014244 012746 013046    MOV      #FRMT11, -(SP)
957 014250 012746 000001    MOV      @1, -(SP)
958 014254 010600          MOV      SP, R0
959 014256 104017          EMT      C$PNTF
960 014260 062706 000004    ADD      #4, SP
961 014264 004737 012042    JSR      PC, LINE1
962 014270          DODU      UNITST          ;DROP THE UNIT
963 014270 013700 002124    MOV      UNITST, R0
964 014274 104053          EMT      C$DODU
965 014276          DOCLN
966 014276 104044          EMT      C$DCLN
967 014300          99$:
968 014300 000205          RTS      R5
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992 014302 005037 002156    CHERR: CLR      DERFLG          ;CLEAR OUT DRIVE ERROR FLAG
993 014306 032737 176000 002160    BIT      #176000, E.CS      ;ANY ERRORS SET
994 014314 001001          BNE      199$          ;IF YES, INVESTIGATE
995 014316 000205          RTS      R5            ;NO, EXIT
996 014320 023727 002262 000004 199$: CMP      TMPFNC, #GSTAT    ;FUNCTION-NOP, RESET, GETSTATUS
997 014326 002401          BLT      98$          ;YES, GO CHECK IF ONLY DRIVE ERROR
998 014330 000414          BR       1$          ;YES SERVICE ERROR
999 014332 023727 002262 000002 98$: CMP      TMPFNC, #WRCHK    ;
1000 014340 001410          BEQ      1$          ;GET E.CS
1001 014342 013700 002160    MOV      E.CS, R0
1002 014346 042700 001777    BIC      #1777, R0

```

.SBTTL ROUTINE TO CHECK FOR CONTROLLER ERRORS

```

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

```


CVRLAA.P11

14-APR-78 15:04

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```

1003 014352 022700 140000      CMP      #140000,R0      ;DRIVE ERROR ALONE?
1004 014356 001001      BNE      1$      ;NO, GO SERVICE
1005 014360 000205      RTS      R5      ;YES, EXIT
1006
1007 014362 012701 011161      1$:      MOV      #EM102,R1      ;GET START OF STRING
1008 014366 005737 002160      TST      E.CS      ;IS COMPOSITE ERROR SET?(BETTER BE)
1009 014372 100003      BPL      99$      ;IT'S NOT SOMETHING IS WRONG
1010 014374 004537 015204      JSR      R5,FIX      ;YES, PUT "COMP" IN STRING
1011 014400 005700      COMP      ;"COMP"
1012 014402 032737 040000 002160 99$:      BIT      #DERR,E.CS      ;DRIVE ERROR SET?
1013 014410 001405      BEQ      3$      ;NO, CONTINUE
1014 014412 005237 002156      INC      DERFLG      ;SET DRV ERROR FLAG
1015 014416 004537 015204      JSR      R5,FIX      ;YES, PUT "DRV" INTO STRING
1016 014422 005541      DEMES      ;"DRV"
1017 014424 032737 020000 002160 3$:      BIT      #NXM,E.CS      ;NON-EXISTENT MEMORY ERROR?
1018 014432 001403      BEQ      4$      ;NO, CONTINUE
1019 014434 004537 015204      JSR      R5,FIX      ;YES, PUT "NXM" INTO STRING
1020 014440 005546      NXMMES      ;"NXM"
1021 014442 032737 002000 002160 4$:      BIT      #OPI,E.CS      ;IS OPI SET?
1022 014450 001422      BEQ      6$      ;NO, GO CHECK BITS 11 & 12
1023 014452 004537 015204      JSR      R5,FIX      ;PUT "OPI" INTO STRING
1024 014456 005553      OPIMES      ;"OPI"
1025 014460 032737 004000 002160      BIT      #BIT11,E.CS      ;HEADERCRC ERROR?
1026 014466 001403      BEQ      5$      ;NO, GO CHECK HEADER NOT FOUND
1027 014470 004537 015204      JSR      R5,FIX      ;GO PUT "HCRC" IN STRING
1028 014474 005560      HCRMES      ;"HCRC"
1029 014476 032737 010000 002160 5$:      BIT      #BIT12,E.CS      ;HEADER NOT FOUND?
1030 014504 001422      BEQ      8$      ;NO, GO PUT "CRLF" IN STRING
1031 014506 004537 015204      JSR      R5,FIX      ;PUT "HNF" IN STRING
1032 014512 005566      HNFMES      ;"HNF"
1033 014514 000416      BR      8$      ;PUT "CRLF" IN STRING
1034 014516 032737 004000 002160 6$:      BIT      #BIT11,E.CS      ;DATA CRC ERROR?
1035 014524 001403      BEQ      7$      ;NO, GO CHECK DATA LATE
1036 014526 004537 015204      JSR      R5,FIX      ;PUT "DCK" IN STRING
1037 014532 005573      DCKMES      ;"DCK"
1038 014534 032737 010000 002160 7$:      BIT      #BIT12,E.CS      ;DATA LATE ERROR?
1039 014542 001403      BEQ      8$      ;NO, GO PUT IN "CRLF"
1040 014544 004537 015204      JSR      R5,FIX      ;PUT "DLT" IN STRING
1041 014550 005600      DLTMES      ;"DLT"
1042 014552 004537 015204      8$:      JSR      R5,FIX
1043 014556 005673      MSCRLF      ;
1044 014560 004537 015204      JSR      R5,FIX
1045 014564 000000      RESTMS:  .WORD      0      ;HEADER FROM TEST
1046 014566 105011      CLRB      (R1)      ;PUT TERMINATOR IN
1047
1048 014570      ERRDF      300,LF,ERR6
1049 014570 104462      TRAP      T$ERRCODE
1050 014572 000454      .WORD      300
1051 014574 005676      .WORD      LF
1052 014576 011626      .WORD      ERR6
1053
1054 014600 000205      RTS      R5      ;EXIT ROUTINE
1055
1056 *****
1057 ;* ROUTINE TO LOAD RLCS WITH FUNCTION TO BE PERFORMED FOR RL11
1058 ;*      CALL:      JSR      R5,LOFUNC

```

CVRLAA.P11

14-APR-78 15:04

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```

1059      ;*
1060      ;*
1061      ;*
1062      ;*
1063      ;*
1064      014602 012537 002200      LDFUNC: MOV      (R5)+,LDCSR      ;GET BITS TO LOAD
1065      014606 005737 002156      TST      DERFLG
1066      014612 001424      BEQ      98$
1067      014614 013746 002146      MOV      B,CS, -(SP)
1068      014620 012777 000013 165304      MOV      #13,ARLDA
1069      014626 012737 000004 002146      MOV      #GSTAT,B,CS
1070      014634 053737 002144 002146      BIS      DRIVE,B,CS
1071      014642 013777 002146 165256      MOV      B,CS,ARLCS
1072      014650 012637 002146      MOV      (SP)+,B,CS
1073      014654 032777 000200 165244 99$: BIT      #200,ARLCS
1074      014662 001774      BEQ      99$
1075      014664 010346 98$: MOV      R3, -(SP)      ;SAVE R3
1076      014666 042737 177661 002200      BIC      #177661,LDCSR      ;CLEAR ALL BUT FUNC & INTR EN
1077      014674 013737 002200 015020      MOV      LDCSR,FNDFNC      ;SAVE FUNCTION
1078      014702 042737 000100 015020      BIC      #INTEN,FNDFNC      ;ONLY FUNCTION
1079      014710 013737 015020 002262      MOV      FNDFNC,TMPFNC
1080      014716 012703 015022      MOV      #HDRLST,R3      ;GET HEADER LIST
1081      014722 006237 015020      ASR      FNDFNC      ;ALIGN TO RIGHT
1082      014726 001404      BEQ      2$
1083      014730 022323 1$: CMP      (R3)+,(R3)+      ;BUMP R3 BY 4
1084      014732 005337 015020      DEC      FNDFNC      ;FOUND IT
1085      014736 001374      BNE      1$      ;NO,KEEP LOOKING
1086      014740 032737 000100 002200 2$: BIT      #INTEN,LDCSR      ;YES,DO WE WANT FLAG OR INTR
1087      014746 001401      BEQ      3$      ;FLAG BRANCH
1088      014750 005723      TST      (R3)+      ;INTR POINT TO THAT ONE
1089      014752 011303 3$: MOV      (R3),R3      ;SET HEADER
1090      014754 010337 014564      MOV      R3,RESTMS      ;SET UP HEADER
1091      014760 053737 002144 002200      BIS      DRIVE,LDCSR      ;SELECT DRIVE
1092      014766 052737 000200 002200 4$: BIS      #200,LDCSR      ;CONTROLLER READY
1093      014774 013777 002200 165124      MOV      LDCSR,ARLCS
1094      015002 004537 015362      JSR      R5,BEFORE
1095      015006 042777 000200 165112 5$: BIC      #200,ARLCS
1096      015014 012603      MOV      (SP)+,R3      ;RESTORE R3
1097      015016 000205      RTS      R5      ;EXIT
1098
1099      015020 000000      FNDFNC: .WORD      0
1100
1101      015022 005761      HDRLST: NOPMES
1102      015024 006012      NOPINT
1103
1104      ;*****
1105      ;THIS ROUTINE WILL CHECK RLV11 CSR FOR COMP,HNF AND OPI ERRORS
1106      ;IN THE MAINTENANCE FORCED OPI TESTS. IT WILL MERGE THE ERROR PRINTOUT
1107      ;WITH THE TEST ERROR MESSAGE. DRIVE ERRORS WILL BE IGNORED.
1108
1109      ;*****
1110      ;*****
1111      ;*****
1112      015026 010146      CALL      JSR      R5,CHKOPI
1113      015030 012701 112000      CHKOPI: MOV      R1, -(SP)
1114      015034 005037 002252      MOV      #112000,R1      ;EXPECTED RESULTS
1115      CLR      MATFLG      ;CLEAR ERROR FOUND FLAG

```


CVRLAA.P11

14-APR-78 15:04

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```

1115 015040 043701 002160      BIC      E.CS,R1      ;CHECK COMP,HNF,OPI
1116 015044 005701              TST      R1
1117 015046 001001              BNE      1$
1118 015050 000453              BR       6$
1119 015052 012701 011161      1$:    MOV     #EM102,R1      ;EXPECTED ERRORS NOT SET
1120 015056 004537 015204      JSR      R5,FIX      ;ALL EXPECTED ERRORS SET,EXIT
1121 015062 005605              EXPMES      ;GET START OF TEXT STRING
1122 015064 032737 100000 002160  BIT      #BIT15,E.CS      ;STORE MESSAGE
1123 015072 001405              BEQ      2$      ;EXPECTED
1124 015074 005237 002252      INC      MATFLG      ;IS COMP SET?
1125 015100 004537 015204      JSR      R5,FIX      ;NO CONTINUE ERROR SEARCH
1126 015104 005700              COMP      ;YES, SET ERROR FOUND
1127 015106 032737 010000 002160 2$:    BIT      #BIT12,E.CS      ;STORE COMP MESSAGE
1128 015114 001405              BEQ      3$      ;IS HNF SET?
1129 015116 005237 002252      INC      MATFLG      ;NO CONTINUE ERROR SEARCH
1130 015122 004537 015204      JSR      R5,FIX      ;YES, SET ERROR FOUND
1131 015126 005566              HNFMES      ;STORE HNF MESSAGE
1132 015130 032737 002000 002160 3$:    BIT      #BIT10,E.CS      ;IS OPI SET?
1133 015136 001405              BEQ      4$      ;NO COMPLETE MESSAGE
1134 015140 005237 002252      INC      MATFLG      ;YES, SET ERROR FOUND
1135 015144 004537 015204      JSR      R5,FIX      ;STORE OPI MESSAGE
1136 015150 005553              OPIMES      ;CHECK IF EXPECTED ERRORS FOUND
1137 015152 005737 002252      TST      MATFLG
1138 015156 001003              BNE      5$
1139 015160 004537 015204      JSR      R5,FIX      ;STORE MESSAGE
1140 015164 005642              NONMES      ;NO EXPECTED ERRORS FOUND
1141 015166 004537 015204      5$:    JSR      R5,FIX
1142 015172 005673              MSCRLF
1143 015174 105011              CLRB      (R1)      ;STORE MESSAGE TERMINATOR
1144 015176 005725              TST      (R5)+      ;RETURN TO PRINT ERROR
1145 015200 012601              6$:    MOV     (SP)+,R1
1146 015202 000205              RTS      R5
1147
1148 *****
1149 ;ROUTINE TO MOVE ASCII STRINGS
1150 ;USES REGISTERS R1 - WHERE STRING IS BEING BUILT
1151 ;
1152 ;      CALL      JSR      R5,FIX
1153 ;      .WORD      ;ADDRESS OF STRING TO MOVE
1154
1154 015204 012500      FIX:    MOV     (R5)+,R0      ;GET ADDRESS AND MOVE RETURN
1155 015206 112021      1$:    MOVB    (R0)+,(R1)+      ;GET BYTE AND UPDATE
1156 015210 001376              BNE      1$      ;WATCH 0 BYTE TERMINATOR
1157 015212 105741              TSTB     -(R1)      ;BACK UP OVER ZERO BYTE
1158 015214 000205              RTS      R5      ;EXIT
1159
1160 *****
1161 ;RLV11 MAINTENANCE SUBROUTINE FOR CRC CALCULATIONS
1162 ;ROUTINE TO RETRIEVE PATTERN AND CALCULATE CRC OF PATTERN+3
1163 ;AND CRC OF CRC OF PATTERN+4.
1164 ;CRC OF PATTERN+3 WILL BE STORED IN "GDCRCA".
1165 ;CRC OF CRC OF PATTERN+4 WILL BE STORED IN "GDCRCB".
1166 ;PATTERN WILL BE STORED IN "GDDATA".
1167 ;
1168 ;      CALL      JSR      R5,CALCRC
1169 ;      .WORD      ;PATTERN IN DA
1170

```

CVRLAA.P11

14-APR-78 15:04

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```

1171
1172 015216 012537 002240          CALCRC: MOV      (R5)+,GCRCP1    ;STORE PATTERN
1173 015222 013737 002240 002222      MOV      GCRCP1,TEMP1
1174 015230 013737 002222 002220      MOV      TEMP1,TEMP5
1175 015236 062737 000003 002220      ADD      #3,TEMP5      ;ADD 3 TO PATTERN
1176 015244 013737 002220 002222      MOV      TEMP5,TEMP1
1177 015252 013737 002222 015266      MOV      TEMP1,15
1178 015260 004537 016112          JSR      R5,SIMBCC      ;CALCULATE EXPECTED CRC
1179 015264 000020          16.      ;DATA BITS
1180 015266 000000          15:      ;INITIAL PATTERN+3
1181 015270 000000          .WORD    0
1182 015272 013737 002210 002242      MOV      CALBCC,GDCRCA    ;SAVE CRC OF PATTERN+3
1183 015300 005237 002220          INC      TEMP5      ;VALUE=PATTERN+4
1184 015304 013737 002220 002222      MOV      TEMP5,TEMP1
1185 015312 013737 002222 015326      MOV      TEMP1,25
1186 015320 004537 016112          JSR      R5,SIMBCC      ;CALCULATE EXPECTED CRC
1187 015324 000020          16.      ;DATA BITS
1188 015326 000000          25:      ;INITIAL PATTERN+4
1189 015330 000000          .WORD    0
1190 015332 013737 002210 015346      MOV      CALBCC,35      ;STORE CRC FOR NEXT CALL
1191 015340 004537 016112          JSR      R5,SIMBCC      ;CAL. CRC OF CRC OF DA+4
1192 015344 000020          16.      ;DATA BITS
1193 015346 000000          35:      ;CRC OF DA+4
1194 015350 000000          .WORD    0
1195 015352 013737 002210 002244      MOV      CALBCC,GDCRCB    ;STARTING CRC=0
1196 015360 000205          RTS      R5      ;SAVE CRC OF CRC OF DA+4
1197
1198      ;LOAD REGISTERS BEFORE FUNCTION
1199      ;CALL: JSR      R5,BEFORE
1200
1201 015362 017737 164540 002146      BEFORE: MOV      @RLCS,B.CS      ;READ CS
1202 015370 017737 164534 002150      MOV      @RLSA,B.BA      ;READ BA
1203 015376 017737 164530 002152      MOV      @RLDA,B.DA      ;READ DA
1204 015404 017737 164524 002154      MOV      @RLMP,B.MP      ;READ MP
1205 015412 000205          RTS      R5
1206
1207
1208      ;LOAD REGISTERS AT ERROR
1209      ;CALL: JSR      R5,AFTER
1210
1211 015414 017737 164506 002160      AFTER: MOV      @RLCS,E.CS      ;READ CS
1212 015422 017737 164502 002162      MOV      @RLBA,E.BA      ;READ BA
1213 015430 017737 164476 002164      MOV      @RLDA,E.DA      ;READ DA
1214 015436 017737 164472 002166      MOV      @RLMP,E.MP      ;READ MP
1215 015444 017737 164464 002170      MOV      @RLMP,E.MP1     ;READ MP
1216 015452 000205          RTS      R5
1217
1218      ;ROUTINE TO SETUP BUFFERS FOR RLVI1 MAINTENANCE FUNCTION
1219      ;BUF1 IS SET WITH 256 WORDS OF PATTERN
1220      ;BUF2 IS CLEARED BEFORE MAINTENANCE FUNCTION
1221      CALL      JSR      R5,SETPAT
1222      .WORD      ;PATTERN FOR BUFFER
1223
1224 015454 010146          SETPAT: MOV      R1,-(SP)
1225 015456 010246          MOV      R2,-(SP)
1226 015460 012537 002246          MOV      (R5)+,GDDATP

```


CVRLAA.P11 14-APR-78 15:04

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```

1227 015464 012701 003500      MOV      #BUF1,R1      ;FIRST BUFFER START
1228 015470 012702 000400      MOV      #256,R2
1229 015474 013721 002246 1$:  MOV      GDDATP,(R1)+
1230 015500 005302      DEC      R2
1231 015502 001374      BNE      1$      ;STORE PATTERN IN 256 WORDS
1232 015504 012701 004500      MOV      #BUF2,R1      ;START OF SECOND BUFFER
1233 015510 012702 000377      MOV      #255,R2
1234 015514 005021 2$:  CLR      (R1)+
1235 015516 005302      DEC      R2
1236 015520 001375      BNE      2$      ;CLEAR 255 WORDS OF SECOND BUFFER
1237 015522 012721 123456      MOV      #123456,(R1)+ ;STORE IN LAST BUFFER WORD
1238 015526 012602      MOV      (SP)+,R2
1239 015530 012601      MOV      (SP)+,R1
1240 015532 000205      RTS      R5

```

```

;ROUTINE TO LOAD RLCS WITH RLV11 MAINT. FUNCTION
;EITHER FLAG DRIVEN OR INTERRUPT MODE.
;
CALL      JSR      R5,LDGUN

```

```

1245 015534 000000      .WORD      ;MAINT!INTEN
1246 015536 000000      .WORD      ;WORD COUNT COMP.
1247 015540 000000      .WORD      ;MAINTENANCE MESSAGE
1248
1249 015542 012537 002200 LDGUN: MOV      (R5)+,LDCSR      ;GET FUNCTION
1250 015546 012577 164362      MOV      (R5)+,JRLMP      ;LOAD WORD COUNT
1251 015552 012537 014564      MOV      (R5)+,RESTMS      ;GET MESSAGE
1252 015556 005037 002262      CLR      TMPFNC      ;CLEAR FUNCTION STORAGE
1253 015562 012777 003500 164340      MOV      #BUF1,JRLBA      ;SET BA REGISTER
1254 015570 013777 002240 164334      MOV      GCRCPT,JRLDA      ;LOAD DA REGISTER
1255 015576 042737 177661 002200      BIC      #177661,LDCSR      ;CLEAR ALL BUT FUNC.+INT.
1256 015604 053737 002144 002200      BIS      DRIVE,LDCSR      ;SELECT DRIVE
1257 015612 052737 000200 002200      BIS      #200,LDCSR      ;CONTROLLER READY
1258 015620 013777 002200 164300      MOV      LDCSR,JRLCS      ;LOAD CS REGISTER
1259 015626 004537 015362      JSR      R5,BEFORE      ;STORE REGISTERS BEFORE OPERATION
1260 015632 042777 000200 164266      BIC      #200,JRLCS      ;CLEAR CONTROLLER READY
1261 015640 000205      RTS      R5      ;RETURN

```

```

;ROUTINE TO SETUP COMPLEMENT BUFFERS FOR RLV11 MAINTENANCE FUNCTION
;BUF1 IS SET WITH PATTERN
;BUF1+1 IS SET WITH COMPLEMENT OF PATTERN

```

```

1267 1268 1269      CALL      JSR      R5,SETCMP      ;PATTERN FOR BUFFER
1270      .WORD
1271
1272 015642 010146 002246 SETCMP: MOV      R1,-(SP)
1273 015646 012537 003500      MOV      R2,-(SP)
1274 015652 012701 000400      MOV      (R5)+,GDDATP
1275 015662 013737 002246 002250      MOV      #BUF1,R1      ;FIRST BUFFER START
1276 015670 013721 002250 1$:  MOV      #256,R2      ;BUFFER COUNT
1277 015674 005137 002250      MOV      GDDATP,GDATMP      ;STORE DATA PATTERN FOR BUF FILL
1278 015700 005302      COM      GDDATMP,(R1)+ ;STORE COMP. IN NEXT BUF LOCATION
1279 015702 001372      DEC      R2
1280 015704 012701 004500      BNE      1$      ;CHECK FOR BUFFER END
1281 015710 012702 000377      MOV      #BUF2,R1      ;SETUP TO CLEAR BUF2
1282 015714 005021 2$:  CLR      #255,R2

```

CVRLAA.P11 14-APR-78 15:04

ROUTINE TO CHECK FOR CONTROLLER ERRORS

1283 015716 005302
 1284 015720 001375
 1285 015722 012721 123456
 1286 015726 012602
 1287 015730 012601
 1288 015732 000205
 1289
 1290
 1291
 1292
 1293
 1294
 1295

```

DEC      R2
BNE      2$      ;CHECK FOR BUF2 END
MOV      #123456,(R1)+ ;STORE IN LAST BUFFER WORD
MOV      (SP)+,R2
MOV      (SP)+,R1
RTS      R5

```

ROUTINE TO SETUP BUFFER WITH RANDOM NUMBERS FOR RLV11 MAINT. FUNCTION
 SAME PATTERN IS USED FOR EACH CONTROLLER
 END OF PASS WILL CHANGE RANDOM PATTERN PRIMES
 CALL JSR R5,SETRAN

1296 015734 010146
 1297 015736 010246
 1298 015740 012701 003500
 1299 015744 012702 000400
 1300 015750 004537 016014
 1301 015754 013721 002272
 1302 015760 005302
 1303 015762 001372
 1304 015764 012701 004500
 1305 015770 012702 000377
 1306 015774 005021
 1307 015776 005302
 1308 016000 001375
 1309 016002 012721 123456
 1310 016006 012602
 1311 016010 012601
 1312 016012 000205
 1313
 1314
 1315
 1316
 1317
 1318
 1319
 1320

```

SETRAN: MOV      R1,-(SP)
          MOV      R2,-(SP)
          MOV      #BUF1,R1      ;FIRST BUFFER START
          MOV      #256,R2      ;BUFFER COUNT
1$:      JSR      R5,RAND      ;GET RANDOM NUMBER
          MOV      LONUM,(R1)+ ;STORE IN BUFFER
          DEC      R2          ;CHECK FOR BUFFER END
          BNE      1$
          MOV      #BUF2,R1      ;SETUP TO CLEAR BUF2
          MOV      #255,R2
2$:      CLR      (R1)+
          DEC      R2
          BNE      2$      ;CHECK FOR BUFFER END
          MOV      #123456,(R1)+ ;STORE IN LAST BUFFER WORD
          MOV      (SP)+,R2
          MOV      (SP)+,R1
          RTS      R5

```

THIS ROUTINE IS A DOUBLE PRECISION PSEUDO RANDOM NUMBER GENERATOR
 WITH A RANGE OF 0 TO 2(+33)-1.

1321 016014 010146
 1322 016016 010246
 1323 016020 010346
 1324 016022 013703 002272
 1325 016026 013701 002270
 1326 016032 012702 177771
 1327 016036 006303
 1328 016040 006101
 1329 016042 005202
 1330 016044 001374
 1331 016046 063703 002272
 1332 016052 005501
 1333 016054 063701 002270
 1334 016060 062703 001057
 1335 016064 005501
 1336 016066 062701 047401
 1337 016072 010337 002272
 1338 016076 010137 002270

```

CALL:    JSR      R5,RAND      ;CALL THE ROUTINE
          RETURN              ;RETURN HERE THE RANDOM NUMBER
          ;WILL BE IN HINUM,LONUM
RAND:    MOV      R1,-(SP)
          MOV      R2,-(SP)
          MOV      R3,-(SP)
          MOV      LONUM,R3
          MOV      HINUM,R1
          MOV      #-7,R2
1$:      ASL      R3
          ROL      R1
          INC      R2
          BNE      1$
          ADD      LONUM,R3
          ADC      R1
          ADD      HINUM,R1
          ADD      #1057,R3
          ADC      R1
          ADD      #47401,R1
          MOV      R3,LONUM
          MOV      R1,HINUM
          ;CALL THE ROUTINE
          ;RETURN HERE THE RANDOM NUMBER
          ;WILL BE IN HINUM,LONUM
          ;PUSH R1 ON STACK
          ;PUSH R2 ON STACK
          ;PUSH R3 ON STACK
          ;SET R3 WITH LOW
          ;SET R1 WITH HIGH
          ;SET SHIFT COUNTER
          ;SHIFT R3 LEFT AND
          ;ROTATE CARRY INTO R1 AND
          ;CHECK FOR DONE
          ;CONTINUE SHIFT LOOP
          ;ADD NUMBER TO MAKE X 129
          ;PROPOGATE CARRY
          ;ADD NUMBER TO MAKE X 129
          ;ADD LOW CONSTANT
          ;PROPOGATE CARRY
          ;ADD HIGH CONSTANT
          ;SAVE R3
          ;SAVE R1

```


CVRLAA.P11 14-APR-78 15:04

ROUTINE TO CHECK FOR CONTROLLER ERRORS

```

1339 016102 012603      MOV      (SP)+,R3      ;POP STACK INTO R3
1340 016104 012602      MOV      (SP)+,R2      ;POP STACK INTO R2
1341 016106 012601      MOV      (SP)+,R1      ;POP STACK INTO R1
1342 016110 000205      RTS      R5              ;RETURN

```

.SBTTL ROUTINE TO CALCULATE CRC

```

;ROUTINE WILL CALCULATE A CRC-16 CRC ON A WORD OF
;1-16 BITS IN LENGTH, RESULT IS RETURNED IN "CALBCC"

```

```

:
:      CALL:   JSR      R5,SIMBCC
:              .WORD    ;NUMBER OF BITS (1-16)
:              .WORD    ;DATA FOR CRC CALCULATION
:              .WORD    ;PREVIOUS OR STARTING CRC
:              ;(SHOULD BE ZEROED FOR START)
:
:      ROUTINE USES R0,R1,R2
:
SIMBCC: MOV      R0,-(SP)      ;SAVE R0
        MOV      R1,-(SP)      ;SAVE R1
        MOV      R2,-(SP)      ;SAVE R2
        MOV      (R5)+,TEMP2    ;GET NUMBER OF BITS
        MOV      (R5)+,TEMP3    ;GET DATA FOR CRC CALCULATION
        MOV      (R5)+,TEMP4    ;GET STARTING CRC
1$:     CLR      BCCFBK          ;
        MOV      TEMP4,R0        ;GET PRESENT CRC
        ROR      TEMP3          ;ROTATE NEW DATA
        ADC      R0             ;MERGE NEW WITH OLD
        BIT      #1,R0          ;BIT 0 SET
        BEQ      2$            ;IF NOT CONTINUE
        COM      BCCFBK          ;
2$:     MOV      XPOLY,R0        ;GET CRC POLYNOMIAL (CRC-16)
        COM      R0             ;COMPLIMENT POLYNOMIAL
        ;CLEAR CARRY
        MOV      TEMP4,BCCFBK,R0
        MOV      TEMP4,R1
        MOV      R1,R2
        BIC      R1,R0
        BIC      BCCFBK,R2
        BIS      R2,R0
        BIC      XPOLY,TEMP4
        BIS      R0,TEMP4
        DEC      TEMP2
        BNE      1$
        MOV      TEMP4,CALBCC
        MOV      (SP)+,R2
        MOV      (SP)+,R1
        MOV      (SP)+,R0
        RTS      R5              ;RETURN

```

```

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

```

```

1339 016102 012603
1340 016104 012602
1341 016106 012601
1342 016110 000205
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357 016112 010046
1358 016114 010146
1359 016116 010246
1360 016120 012537 002212
1361 016124 012537 002214
1362 016130 012537 002216
1363 016134 005037 002206
1364 016140 013700 002216
1365 016144 006037 002214
1366 016150 005500
1367 016152 032700 000001
1368 016156 001402
1369 016160 005137 002206
1370 016164 013700 002202
1371 016170 005100
1372 016172 040037 002206
1373 016176 000241
1374 016200 006037 002216
1375 016204 013700 002206
1376 016210 013701 002216
1377 016214 010102
1378 016216 040100
1379 016220 043702 002206
1380 016224 050200
1381 016226 043737 002202 002216
1382 016234 050037 002216
1383 016240 005337 002212
1384 016244 001333
1385 016246 013737 002216 002210
1386 016254 012602
1387 016256 012601
1388 016260 012600
1389 016262 000205
1390
1391
1392
1393
1394

```

CVRLAA.P11 14-APR-78 15:04

ROUTINE TO CALCULATE CRC

```

1395
1396
1397 016264 005237 002174      TRPHAN: INC      TRPFLG      ;INDICATE TRAP
1398 016270 000002              RTI              ;RETURN
1399
1400 016272              BGNSRV
1401
1402 016272 005237 002176      INTSRV: INC      INTFLG      ;INDICATE INTERRUPT
1403
1404 016276              ENDSRV
1405 016276              L10020:
1406 016276 000002              RTI
1407
1408      ;ROUTINE USED IN TIMING OPI
1409 016300 005237 002176      TIMSRV: INC      INTFLG
1410 016304              ABORTWAIT
1411 016304 104021              EMT      CSABRT
1412 016306 000002              RTI
1413
1414
1415      ;ROUTINE TO WAIT FOR DRIVE READY
1416 016310 010146              WTDORDY: MOV      R1, -(SP)      ;SAVE R1
1417 016312 012701 003720      MOV      #2000, R1      ;TIME OUT OF 200 MILLISECONDS
1418 016316 032777 000001 163602 1$: BIT      #DRDY, DRLCS      ;DRIVE READY?
1419 016324 001011              BNE      2$      ;YES, EXIT
1420
1421 016326              WAITUS      #1      ;WAIT A WHILE
1422 016326 012700 000001      MOV      #1, R0
1423 016332 104027              EMT      CSWTU
1424 016334 005301              DEC      R1
1425 016336 001367              BNE      1$      ;CHECK IF TIME UP
1426                                     ;NO, GO CHECK DRIVE READY
1427 016340              ERRDF      200, DRTIM, ERR5 ;DRIVE READY DID NOT SET
1428 016340 104462              TRAP      T$ERRCODE
1429 016342 000310              .WORD      200
1430 016344 006267              .WORD      DRTIM
1431 016346 011614              .WORD      ERR5
1432
1433 016350 012601              2$: MOV      (SP)+, R1      ;RESTORE
1434 016352 000205              RTS      R5      ;EXIT
1435
1436      ;ROUTINE TO WAIT FOR CONTROLLER READY
1437 016354 010146              WTCRDY: MOV      R1, -(SP)      ;SAVE R1
1438 016356 012701 017500      MOV      #8000, R1      ;WAIT 800 MILLISECONDS
1439 016362 032777 000200 163536 1$: BIT      #CRDY, DRLCS      ;CONTROLLER READY
1440 016370 001014              BNE      2$      ;YES, EXIT
1441 016372              WAITUS      #1      ;WAIT A WHILE
1442 016372 012700 000001      MOV      #1, R0
1443 016376 104027              EMT      CSWTU
1444 016400 005301              DEC      R1
1445 016402 001367              BNE      1$      ;CHECK IF TIME UP
1446                                     ;NO GO BACK
1447 016404 004537 015414      JSR      R5, AFTER      ;GET REGISTERS
1448
1449 016410              ERRDF      100, CRTIM, ERR6 ;CONTROLLER TIMED OUT
1450 016410 104462              TRAP      T$ERRCODE

```


CVRLAA.P11 14-APR-78 15:04

ROUTINE TO CALCULATE CRC

```

1451 016412 000144      .WORD 100
1452 016414 006242      .WORD CRTIM
1453 016416 011626      .WORD ERR6
1454
1455 016420 000402      BR 3$ ;EXIT
1456
1457 016422 004537 015414 2$: JSR RS, AFTER ;GET REGISTERS
1458 016426 012601      3$: MOV (SP)+, R1
1459 016430 000205      RTS RS ;EXIT
1460
1461
1462
1463 016432      ENDMOD
1464
1465
1466
1467
1468
1469 016432      .SBTTL **TEST 1** - RLCS WRITE ADDRESSABILITY
1470 016432      BGNTST ;****START OF TEST****
1471      STARS
1472      ;*****
1473      ;TEST TO SEE IF WE CAN ADDRESS THE CONTROL
1474      ;AND STATUS REGISTER. IF WE TRAP WE WILL REPORT
1475      ;THE ERROR AND ABORT. AFTER THIS TEST WE ONLY KNOW
1476      ;THAT WE CAN ADDRESS THE REGISTER.
1477 016432      STARS
1478      ;*****
1479
1480 016432 005037 002174 1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
1481 016436      2$: SETVEC ERRVEC, #TRPHAN, #340 ;SET TO CATCH TRAP
1482 016436 012746 000340      MOV #340, -(SP)
1483 016442 012746 016264      MOV #TRPHAN, -(SP)
1484 016446 013746 002204      MOV ERRVEC, -(SP)
1485 016452 012746 000003      MOV #3, -(SP)
1486 016456 104037      EMT C$SVEC
1487 016460 062706 000010      ADD #10, SP
1488
1489 016464 012777 177777 163434      MOV #177777, @RLCS ;ADDRESS RLCS
1490 016472      CLRVEC ERRVEC ;RELEASE TRAP VECTOR
1491 016472 013700 002204      MOV ERRVEC, R0
1492 016476 104036      EMT C$CVEC
1493 016500 005737 002174      TST TRPFLG ;TRAP OCCURRED???
1494 016504 001407      BEQ 3$ ;NO, OKAY PROCEED
1495 016506 013737 002126 002234      MOV RLCS, GDDAT ;SET UP ERROR DATA
1496
1497 016514      ERRSF 0, EM1, ERR1 ;BUS TIMEOUT IN ADDRESSING RLCS
1498 016514 104461      TRAP T$ERRCODE
1499 016516 000000      .WORD 0
1500 016520 006315      .WORD EM1
1501 016522 011370      .WORD ERR1
1502 016524      3$: CKLOOP ;CHECK IF /FL:LOE IS SET
1503 016524 104006      EMT C$CLP1
1504 016526      ENDTST ;****END OF TEST****
1505 016526      L10021:
1506 016526 104001      EMT C$SETST

```

CVRLAA.P11 14-APR-78 15:04

TEST 1 - RLCS WRITE ADDRESSABILITY

.SBTTL **TEST 2** - RLBA WRITE ADDRESSABILITY

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

STARS

```

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

```

```

1S: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2S: SETVEC ERRVEC, #TRPHAN, #340 ;SET TO CATCH TRAP
MOV #340, -(SP)
MOV #TRPHAN, -(SP)
MOV ERRVEC, -(SP)
MOV #3, -(SP)
EMT C$SVEC
ADD #10, SP

MOV #177777, @RLBA ;ADDRESS RLBA
CLRVEC ERRVEC ;RELEASE TRAP VECTOR
MOV ERRVEC, R0
EMT C$CVEC
TST TRPFLG ;TRAP OCCURRED???
BEQ 3S ;NO, CONTINUE
MOV RLBA, GDDAT ;SETUP ERROR DATA

ERRSF 1, EM2, ERR1 ;BUS TIMEOUT IN ADDRESSING RLBA
TRAP 1, $ERRCODE
.WORD 1
.WORD EM2
.WORD ERR1
3S: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT C$CLP1
ENDTST ;****END OF TEST****
L10022: EMT C$SETST

```

.SBTTL **TEST 3** - RLDA WRITE ADDRESSABILITY

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

STARS

```

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

```


CVRLAA.P11 14-APR-78 15:04

TEST 3 - RLDA WRITE ADDRESSABILITY

```

1563
1564
1565 016626 005037 002174
1566 016632
1567 016632 012746 000340
1568 016636 012746 016264
1569 016642 013746 002204
1570 016646 012746 000003
1571 016652 104037
1572 016654 062706 000010
1573
1574 016660 012777 177777 163244
1575 016666
1576 016666 013700 002204
1577 016672 104036
1578 016674 005737 002174
1579 016700 001407
1580
1581 016702 013737 002132 002234
1582 016710
1583 016710 104461
1584 016712 000002
1585 016714 006367
1586 016716 011370
1587 016720
1588 016720 104006
1589 016722
1590 016722
1591 016722 104001
1592
1593
1594
1595
1596 016724
1597 016724
1598
1599
1600
1601
1602
1603 016724
1604
1605
1606
1607 016724 005037 002174
1608 016730
1609 016730 012746 000340
1610 016734 012746 016264
1611 016740 013746 002204
1612 016744 012746 000003
1613 016750 104037
1614 016752 062706 000010
1615
1616 016756 012777 177777 163150
1617 016764
1618 016764 013700 002204

```

```

1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2$: SETVEC ERRVEC,TRPHAN,#340 ;SET TO CATCH TRAP
MOV #340,-(SP)
MOV TRPHAN,-(SP)
MOV ERRVEC,-(SP)
MOV #3,-(SP)
EMT CSSVEC
ADD #10,SP

MOV #177777,3RLDA ;ADDRESS RLDA
CLAVEC ERRVEC ;RELEASE TRAP VECTOR
MOV ERRVEC,RO
EMT CSSVEC
TST TRPFLG ;TRAP OCCURRED???
BEQ 3$ ;NO, CONTINUE

MOV RLDA,GDDAT ;SETUP ERROR INFO
ERRSF 2,EM3,ERR1 ;BUS TIMEOUT IN ADDRESSING RLDA
TRAP T$ERRCODE
WORD 2
WORD EM3
WORD ERR1

3$: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT C$CLP1
ENDTST ;****END OF TEST****
L10023: EMT C$SETST

```

.SBTTL **TEST 4** - RLMP WRITE ADDRESSABILITY

```

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

```

```

1607 016724 005037 002174
1608 016730
1609 016730 012746 000340
1610 016734 012746 016264
1611 016740 013746 002204
1612 016744 012746 000003
1613 016750 104037
1614 016752 062706 000010
1615
1616 016756 012777 177777 163150
1617 016764
1618 016764 013700 002204

```

```

1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2$: SETVEC ERRVEC,TRPHAN,#340 ;SET UP TO CATCH TRAP
MOV #340,-(SP)
MOV TRPHAN,-(SP)
MOV ERRVEC,-(SP)
MOV #3,-(SP)
EMT CSSVEC
ADD #10,SP

MOV #177777,3RLMP ;ADDRESS RLMP
CLAVEC ERRVEC ;RELEASE TRAP VECTOR
MOV ERRVEC,RO

```

CVRLAA.P11 14-APR-78 15:04

TEST 4 - RLMP WRITE ADDRESSABILITY

```

1619 016770 104036      EMT      CSCVEC
1620 016772 005737 002174  TST      TRPFLG      ;TRAP OCCURRED???
1621 016776 001407      BEQ      3$          ;NO, CONTINUE
1622 017000 013737 002134 002234  MOV      RLMP,GDDAT  ;SET UP ERROR INFO
1623
1624 017006      ERRSF      3, EM4,ERR1  ;BUS TIMEOUT IN ADDRESSING RLMP
1625 017006 104461      TRAP      T$ERRCODE
1626 017010 000003      .WORD      3
1627 017012 006414      .WORD      EM4
1628 017014 011370      .WORD      ERR1
1629 017016      3$:      CKLOOP      ;CHECK IF /FL:LOE IS SET
1630 017016 104006      EMT      CSCLP1
1631 017020      ENDTST
1632 017020      L10024:      ;****END OF TEST****
1633 017020 104001      EMT      CSETST
1634

```

.SBTTL **TEST 5** - RLCS READ ADDRESSABILITY

```

1635
1636
1637 017022      BGNTST      ;****START OF TEST****
1638 017022      STARS
1639
1640      ;*****
1641      ;TEST TO SEE IF WE CAN ADDRESS THE CONTROL
1642      ;AND STATUS REGISTER. IF WE TRAP WE WILL REPORT
1643      ;THE ERROR AND ABORT. AFTER THIS TEST WE ONLY KNOW
1644      ;THAT WE CAN ADDRESS THE REGISTER.
1645      STARS
1646      ;*****
1647
1648 017022 005037 002174  1$:      CLR      TRPFLG      ;CLEAR TRAP OCCURANCE
1649 017026      2$:      SETVEC      ERRVEC, #TRPHAN, #340 ;SET TO CATCH TRAP
1650 017026 012746 000340  MOV      #340, -(SP)
1651 017032 012746 016264  MOV      #TRPHAN, -(SP)
1652 017036 013746 002204  MOV      ERRVEC, -(SP)
1653 017042 012746 000003  MOV      #3, -(SP)
1654 017046 104037      EMT      CSCVEC
1655 017050 062706 000010  ADD      #10, SP
1656
1657 017054 005777 163046  TST      RLCS      ;ADDRESS RLCS
1658 017060      CLRVEC      ERRVEC      ;RELEASE TRAP VECTOR
1659 017060 013700 002204  MOV      ERRVEC, RO
1660 017064 104036      EMT      CSCVEC
1661 017066 005737 002174  TST      TRPFLG      ;TRAP OCCURRED???
1662 017072 001407      BEQ      3$          ;NO, OKAY PROCEED
1663 017074 013737 002126 002234  MOV      RLCS,GDDAT  ;SET UP ERROR DATA
1664
1665 017102      ERRSF      100, EM1,ERR1  ;BUS TIMEOUT IN ADDRESSING RLCS
1666 017102 104461      TRAP      T$ERRCODE
1667 017104 000144      .WORD      100
1668 017106 006315      .WORD      EM1
1669 017110 011370      .WORD      ERR1
1670 017112      3$:      CKLOOP      ;CHECK IF /FL:LOE IS SET
1671 017112 104006      EMT      CSCLP1
1672 017114      ENDTST
1673 017114      L10025:      ;****END OF TEST****
1674 017114 104001      EMT      CSETST

```


CVRLAA.P11 14-APR-78 15:04

TEST 5 - RLCS READ ADDRESSABILITY

.SBTTL **TEST 6** - RLBA READ ADDRESSABILITY

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

STARS

```

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

```

STARS

```

;*****

```

```

1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2$: SETVEC ERRVEC,STRPHAN,#340 ;SET TO CATCH TRAP
MOV #340,-(SP)
MOV STRPHAN,-(SP)
MOV ERRVEC,-(SP)
MOV #3,-(SP)
EMT CSSVEC
ADD #10,SP

TST JRLBA ;ADDRESS RLBA
CLRVEC ERRVEC ;RELEASE TRAP VECTOR
MOV ERRVEC,RO
EMT CSCVEC
TST TRPFLG ;TRAP OCCURRED???
BEQ 3$ ;NO, CONTINUE
MOV RLBA,GDDAT ;SETUP ERROR DATA

ERRSF 101,EM2,ERR1 ;BUS TIMEOUT IN ADDRESSING RLBA
TRAP TSECODE
WORD 101
WORD EM2
WORD ERR1
3$: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT CSCLP1
ENDTST ;****END OF TEST****
L10026: EMT CSETST

```

.SBTTL **TEST 7** - RLDA READ ADDRESSABILITY

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

STARS

```

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

```

STARS

```

;*****

```

CVRLAA.P11 14-APR-78 15:04

TEST 7 - RLDA READ ADDRESSABILITY

```

1731
1732
1733 017212 005037 002174
1734 017216
1735 017216 012746 000340
1736 017222 012746 016264
1737 017226 013746 002204
1738 017232 012746 000003
1739 017236 104037
1740 017240 062706 000010
1741
1742 017244 005777 162662
1743 017250
1744 017250 013700 002204
1745 017254 104036
1746 017256 005737 002174
1747 017262 001407
1748
1749 017264 013737 002132 002234
1750 017272
1751 017272 104461
1752 017274 000146
1753 017276 006367
1754 017300 011370
1755 017302
1756 017302 104006
1757 017304
1758 017304
1759 017304 104001
1760
1761
1762
1763
1764 017306
1765 017306
1766
1767
1768
1769
1770
1771 017306
1772
1773
1774
1775 017306 005037 002174
1776 017312
1777 017312 012746 000340
1778 017316 012746 016264
1779 017322 013746 002204
1780 017326 012746 000003
1781 017332 104037
1782 017334 062706 000010
1783
1784 017340 005777 162570
1785 017344
1786 017344 013700 002204

```

```

1S: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2S: SETVEC ERRVEC,TRPHAN,#340 ;SET TO CATCH TRAP
MOV #340,-(SP)
MOV TRPHAN,-(SP)
MOV ERRVEC,-(SP)
MOV #3,-(SP)
EMT CSSVEC
ADD #10,SP

TST JRLDA ;ADDRESS RLDA
CLRVEC ERRVEC ;RELEASE TRAP VECTOR
MOV ERRVEC,RO
EMT CSCVEC
TST TRPFLG ;TRAP OCCURRED???
BEQ 3S ;NO, CONTINUE

MOV RLDA,GDDAT ;SETUP ERROR INFO
ERRSF 102,EM3,ERR1 ;BUS TIMEOUT IN ADDRESSING RLDA
TRAP TSECODE
.WORD 102
.WORD EM3
.WORD ERR1
3S: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT CSCLP1 ;****END OF TEST****

ENDTST
L10027: EMT CSETST

```

.SBTTL **TEST 8** - RLMP READ ADDRESSABILITY

```

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

```

```

1S: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2S: SETVEC ERRVEC,TRPHAN,#340 ;SET UP TO CATCH TRAP
MOV #340,-(SP)
MOV TRPHAN,-(SP)
MOV ERRVEC,-(SP)
MOV #3,-(SP)
EMT CSSVEC
ADD #10,SP

TST JRLMP ;ADDRESS RLMP
CLRVEC ERRVEC ;RELEASE TRAP VECTOR
MOV ERRVEC,RO

```


CVRLAA.P11 14-APR-78 15:04

TEST 8 - RLMP READ ADDRESSABILITY

```

1787 017350 104036      EMT      CSCVEC
1788 017352 005737 002174 TST      TRPFLG      ;TRAP OCCURRED???
1789 017356 001407      BEQ      3$      ;NO CONTINUE
1790 017360 013737 002134 002234 MOV      RLMP,GDDAT  ;SET UP ERROR INFO
1791
1792 017366      ERRSF     103, EM4,ERR1 ;BUS TIMEOUT IN ADDRESSING RLMP
1793 017366 104461      TRAP     TSERCODE
1794 017370 000147      .WORD    103
1795 017372 006414      .WORD    EM4
1796 017374 011370      .WORD    ERR1
1797 017376      3$:      CKLOOP
1798 017376 104006      EMT      CSCLP1      ;CHECK IF /FL:LOE IS SET
1799 017400      ENDTST
1800 017400      L10030:
1801 017400 104001      EMT      CSETST      ;****END OF TEST****
1802
1803      .SBTTL  **TEST 9** - BUS RESET OF RLCS
1804
1805 017402      BGNTST      ;****START OF TEST****
1806
1807 017402      STARS
1808      ;*****
1809      ;TEST THAT A BUS RESET WILL CLEAR THE PROPER BITS
1810      ;OF THE CONTROL AND STATUS REGISTER. THOSE BITS ARE
1811      ;BITS 6-1, 8, 9, 10, 11, 12, 13, 15. BIT 15 WILL CLEAR ONLY
1812      ;IF BIT 14 (DRIVE ERROR IS NOT SET). BIT 0 (DRIVE READY)
1813      ;IS A DON'T CARE. IF AT THE START UP THIS TEST BIT
1814      ;14 (DRIVE ERROR) IS SET WE WILL INSIST IF IS THERE AFTER
1815      ;THE "RESET" ALONG WITH BIT 15 (COMPOSITE ERROR). BITS
1816      ;15-10 ARE NOT WRITEABLE.
1817 017402      STARS
1818      ;*****
1819
1820
1821 017402      SETPRI     #PRI07      ;PRIORITY TO SEVEN
1822 017402 012700 000340 MOV      #PRI07,RO
1823 017406 104041      EMT      CSSPRI
1824 017410 012777 000377 162510 MOV      #377,RLCS      ;LOAD ALL RLCS LOADABLE BITS
1825 017416 012737 000200 002234 MOV      #CRDY,GDDAT  ;SETUP EXPECTED
1826 017424 032777 040000 162474 BIT      #DERR,RLCS  ;DRIVE ERR SET?
1827 017432 001403      BEQ      1$      ;IF NOT DON'T EXPECT IT
1828 017434 052737 140000 002234 BIS      #DERR!ERR,GDDAT ;IT'S SET, INIT BETTER NOT CLR
1829 017442 012700 000100      MOV      #100,RO      ;SET UP A WAIT LOOP
1830 017446      BRESET
1831 017446 104033      EMT      CSRESET      ;BUS RESET
1832 017450 005300      DEC      RO      ;WAIT IN CASE OF DRIVE ERROR
1833 017452 001376      BNE      2$
1834 017454 017737 162446 002236 MOV      RLCS,BDDAT  ;READ RLCS
1835 017462 042737 000001 002236 BIC      #DRDY,BDDAT  ;CLEAR OUT DRDY - DON'T CARE
1836 017470 023737 002236 002234 CMP      BDDAT,GDDAT  ;DID INIT WORK
1837 017476 001404      BEQ      3$      ;YES, BRANCH
1838
1839 017500      ERADF     113, EM67,ERR2 ;WRONG DATA IN RLCS
1840 017500 104462      TRAP     TSERCODE
1841 017502 000161      .WORD    113
1842 017504 010525      .WORD    EM67

```

CVRLAA.P11 14-APR-78 15:04

TEST 9 - BUS RESET OF RLCS

.WORD ERR2

3S:
ENDTST
L10031:

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

EMT CSETST

.SBTTL **TEST 10** - BUS RESET OF RLBA

BGNTST

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

STARS

```

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

```

```

MOV      #-2,RLBA      ;SET BA TO ALL 1'S
TST      T.CNTRL        ;RL11??
BEQ      2S             ;NO
BIS      #1,RLBA
CLR      GDDAT          ;CLEAR EXPECTED DATA
BRESET   ;ISSUE BUS INIT
EMT
MOV      RLBA,BDDAT     ;READ RLBA
BEQ      1S             ;IF CLEAR BRANCH

ERRDF    114,EM70,ERR2 ;WRONG DATA IN RLBA
TRAP     T.SERCODE
WORD     114
WORD     EM70
WORD     ERR2

```

1S:

ENDTST
L10032:

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

EMT CSETST

.SBTTL **TEST 11** - BUS RESET OF RLDA

BGNTST

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

STARS

```

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

```

MOV #-1,RLDA ;SET DA TO ALL 1'S

```

1843 017506 011402
1844 017510
1845 017510
1846 017510
1847 017510 104001
1848
1849
1850
1851
1852 017512
1853
1854 017512
1855
1856
1857
1858
1859 017512
1860
1861
1862
1863 017512 012777 177776 162410
1864 017520 005737 002260
1865 017524 001403
1866 017526 052777 000001 162374
1867 017534 005037 002234
1868 017540
1869 017540 104033
1870 017542 017737 162362 002236
1871 017550 001404
1872
1873 017552
1874 017552 104462
1875 017554 000162
1876 017556 010562
1877 017560 011402
1878 017562
1879
1880 017562
1881 017562
1882 017562 104001
1883
1884
1885
1886
1887 017564
1888
1889 017564
1890
1891
1892
1893
1894 017564
1895
1896
1897
1898 017564 012777 177777 162340

```


H05

PAGE: 0059AS
SEQ 0059

CVRLAA.P11 14-APR-78 15:04

TEST 11 - BUS RESET OF RLDA

```

1899 017572 005037 002234          CLR      GDDAT          ;CLEAR EXPECTED
1900 017576          BRESET        ;ISSUE BUS INIT
1901 017576 104033          EMT      CSRESET
1902 017600 017737 162326 002236    MOV      JRLDA,BDDAT      ;READ RLDA
1903 017606 001404          BEQ      1$              ;IF CLEAR BRANCH
1904
1905 017610          ERRDF 11$ EM71,ERR2 ;WRONG DATA IN RLDA
1906 017610 104462          TRAP      T$ERCODE
1907 017612 000163          .WORD    11$
1908 017614 010617          .WORD    EM71
1909 017616 011402          .WORD    ERR2

```

1\$:

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

L10033:

EMT CSETST

.SBTTL **TEST 12** - READ WRITE OF RLCS

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

STARS

```

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

```

```

1931
1932
1933 017622 012703 002700          MOV      #CSPAT,R3      ;SET UP TABLE POINTER OF PATTERNS
1934
1935 017626          BGNSEG
1936 017626 104004          EMT      CSBSEG      ;****START OF SEGMENT****
1937
1938 017630          CSTEST:
1939 017630 011337 002234          MOV      (R3),GDDAT      ;GET PATTERN INTO GDDAT
1940 017634 052737 000200 002234    BIS      #200,GDDAT      ;INSURE GO IS SET
1941 017642 013777 002234 162256    MOV      GDDAT,JRLCS      ;LOAD RLCS (CONTROL AND STATUS)
1942 017650 032777 040000 162250    BIT      #DERR,JRLCS      ;IF DRIVE ERROR PRESENT
1943 017656 001403          BEQ      99$              ;THEN EXPECT DRIVE AND
1944 017660 052737 140000 002234    BIS      #ERR:DERR,GDDAT ;COMPOSITE ERROR
1945 017666 017737 162234 002236    99$: MOV      JRLCS,BDDAT      ;READ RLCS BACK
1946 017674 042737 000001 002236    BIC      #DRDY,BDDAT      ;IGNORE DRIVE READY
1947 017702 023737 002234 002236    CMP      GDDAT,BDDAT      ;DID WE READ WHAT WE LOADED
1948 017710 001404          BEQ      1$              ;YES, THEN BRANCH
1949
1950          ERRDF 4. EMS,ERR2      ;WRONG DATA IN RLCS
1951 017712 104462          TRAP      T$ERCODE
1952 017714 000004          .WORD    4
1953 017716 006441          .WORD    EMS
1954 017720 011402          .WORD    ERR2

```

CVRLAA.P11 14-APR-78 15:04

TEST 12 - READ WRITE OF RLCS

1955 017722
 1956 017722 104010
 1957 017724 000012
 1958
 1959
 1960 017726 005723
 1961 017730 020327 002776
 1962 017734 001335
 1963
 1964 017736
 1965 017736
 1966 017736 104005
 1967 017740
 1968 017740
 1969 017740 104001
 1970
 1971
 1972
 1973
 1974 017742
 1975
 1976 017742
 1977
 1978
 1979
 1980
 1981
 1982 017742
 1983
 1984
 1985
 1986 017742 012703 002304
 1987 017746
 1988 017746 104004
 1989 017750
 1990 017750 011337 002234
 1991 017754 005737 002260
 1992 017760 001403
 1993 017762 042737 000001 002234
 1994 017770 013777 002234 162132
 1995 017776 017737 162126 002236
 1996 020004 023737 002234 002236
 1997 020012 001404
 1998
 1999 020014
 2000 020014 104462
 2001 020016 000005
 2002 020020 006512
 2003 020022 011402
 2004 020024
 2005 020024 104010
 2006 020026 000012

1\$: ESCAPE SEG ; IF /FL:LOE SET LOOP, ELSE EXIT SEG
 EMT C\$ESCAPE
 .WORD 10000\$-

TST (R3)+ ; BUMP FOR NEXT PATTERN
 CMP R3, #CSEND ; CHECK FOR END
 BNE C\$TEST ; NOT END, LOAD NEXT PATTERN

ENDSEG ; *****END OF SEGMENT*****
 10000\$:

ENDTST EMT C\$ESEG ; *****END OF TEST*****
 L10034:

EMT C\$ETST

.SBTTL **TEST 13** - READ WRITE OF RLBA

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

STARS

 : TEST THAT WE CAN WRITE/READ BITS IS THRU 1 OF THE
 : BUS ADDRESS REGISTER. FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
 : GROWING 0 AND SHIFTING 0. BIT 0 IS ALSO LOADED BUT
 : SHOULD ALWAYS COME BACK AS 0
 STARS

BGNSEG MOV #BEGPAT, R3 ; GET START OF PATTERN LIST
 EMT C\$BSEG ; *****START OF SEGMENT*****

BATEST: MOV (R3), GDDAT ; GET PATTERN TO SEND
 TST T.CNTRL ; RL11??
 BEQ 2\$; NO
 BIC #BIT0, GDDAT ; KEEP RLBA EVEN (UNIBUS)
 2\$: MOV GDDAT, JALBA ; LOAD PATTERN TO BUS ADDRESS
 MOV JALBA, BDDAT ; READ IT BACK
 CMP GDDAT, BDDAT ; IS IT CORRECT?
 BEQ 1\$; IF SO, BRANCH

ERRDF 5, EM6, ERR2 ; DATA WRONG IN RLBA
 TRAP T\$ERRCODE

.WORD 5
 .WORD EM6
 .WORD ERR2
 1\$: ESCAPE SEG ; IF /FL:LOE SET LOOP, ELSE EXIT SEG
 EMT C\$ESCAPE
 .WORD 10000\$-

CVRLAA.P11 14-APR-78 15:04

TEST 13 - READ WRITE OF RLBA

```

2007
2008
2009 020030 005723
2010 020032 020327 002512
2011 020036 001344
2012
2013 020040
2014 020040
2015 020040 104005
2016 020042
2017 020042
2018 020042 104001
2019
2020
2021
2022
2023 020044
2024
2025 020044
2026
2027
2028
2029
2030 020044
2031
2032
2033
2034 020044 012703 002304
2035 020050
2036 020050 104004
2037 020052
2038 020052 011337 002234
2039 020056 013777 002234 162046
2040
2041 020064 017737 162042 002236
2042 020072 023737 002234 002236
2043 020100 001404
2044
2045 020102
2046 020102 104462
2047 020104 000006
2048 020106 006540
2049 020110 011402
2050 020112
2051 020112 104010
2052 020114 000012
2053
2054
2055 020116 005723
2056 020120 020327 002512
2057 020124 001352
2058
2059 020126
2060 020126
2061 020126 104005
2062 020130

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

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

EMT CSESEG
;****END OF TEST****

EMT CSETST

.SBTTL **TEST 14** - READ WRITE OF RLDA
BGNTEST
;****START OF TEST****

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

BGNSEG MOV #BEGPAT,R3 ;SET UP POINTER TO PATTERN LIST
;****START OF SEGMENT****
DATEST: EMT CSBSEG

MOV (R3),GDDAT ;GET PATTERN
MOV GDDAT,RDLDA ;LOAD PATTERN IN DA

MOV RDLDA,BDDAT ;READ PATTERN BACK
CMP GDDAT,BDDAT ;IS IT CORRECT?
BEQ IS ;BRANCH IF CORRECT

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

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

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

EMT CSESEG
;****END OF TEST****

ENDTST

```

CVRLAA.P11 14-APR-78 15:04

TEST 14 - READ WRITE OF RLDA

L10036: EMT CSETST

.SBTTL **TEST 15** - BIS OF RLCS

BGNSTST ;****START OF TEST****
STARS:*****
:TEST THAT WE CAN USE THE "BIS" INSTRUCTION ON THE CONTROL
:AND STATUS REGISTER. BITS 8,9 AND 6-1 ARE TESTED TO
:SET INDIVIDUALLY AS WELL AS COLLECTIVELY WITHOUT DESTROYING
:ANY PREVIOUS DATA PATTERN
STARS
:*****BGNSEG MOV #CSPAT,R3 ;GET BEGINNING OF LIST
EMT CSBSEG ;****START OF SEGMENT****1\$: MOV #CRDY,RLCS ;INSURE GO IS THERE
MOV (R3),GDDAT ;SET UP EXPECTED RLCS
BIS #CRDY,GDDAT ;IN GDDAT
BIS (R3),RLCS ;BIT SET PATTERN IN RLCS
BIT #DERR,RLCS ;IF ERROR BIT SET THEN
BEQ 99\$;EXPECT IT ON THE READ
BIS #ERR!DERR,GDDAT ;BACK
99\$: MOV RLCS,BDDAT ;READ RLCS TO CHECK "BIS"
BIC #DRDY,BDDAT ;CLEAR OUT DRIVE READY
CMP BDDAT,GDDAT ;DID BIS WORK?
BEQ 2\$;BRANCH IF OKAY
ERRDF 7,EM61,ERR2 ;WRONG DATA IN RLCS
TRAP T\$ERRCODE
.WORD 7
.WORD EM61
.WORD ERR2
2\$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
EMT C\$ESCAPE
.WORD 10000\$-.

;BIT OR CLEARED OTHER BIT

TST (R3)+ ;GET NEXT PATTERN
CMP #CSEND,R3 ;AT END OF LIST
BNE 1\$;NO GO BACK FOR TEST OF
NEXT PATTERN
;****END OF SEGMENT****ENDSEG
10000\$: EMT C\$ESEG ;****END OF TEST****
ENDTST
L10037: EMT CSETST

.SBTTL **TEST 16** - BIC OF RLCS

2063	020130			
2064	020130	104001		
2065				
2066				
2067				
2068				
2069	020132			
2070	020132			
2071				
2072				
2073				
2074				
2075				
2076	020132			
2077				
2078				
2079				
2080	020132	012703	002700	
2081	020136			
2082	020136	104004		
2083	020140			
2084	020140	012777	000200	161760
2085	020146	011337	002234	
2086	020152	052737	000200	002234
2087	020160	051377	161742	
2088	020164	032777	040000	161734
2089	020172	001403		
2090	020174	052737	140000	002234
2091	020202	017737	161720	002236
2092	020210	042737	000001	002236
2093	020216	023737	002236	002234
2094	020224	001404		
2095				
2096	020226			
2097	020226	104462		
2098	020230	000007		
2099	020232	010051		
2100	020234	011402		
2101	020236			
2102	020236	104010		
2103	020240	000012		
2104				
2105				
2106	020242	005723		
2107	020244	022703	002776	
2108	020250	001333		
2109				
2110	020252			
2111	020252			
2112	020252	104005		
2113	020254			
2114	020254			
2115	020254	104001		
2116				
2117				
2118				

CVRLAA.P11 14-APR-78 15:04

TEST 16 - BIC OF RLCS

```

2119
2120 020256          BGNTST                      ;****START OF TEST****
2121
2122 020256          STARS
2123          ;*****
2124          ;TEST THAT THE "BIC" INSTRUCTION WILL WORK ON THE
2125          ;CONTROL AND STATUS REGISTER.  BITS 8-9 AND 6-1 ARE
2126          ;TESTED.
2127 020256          STARS
2128          ;*****
2129
2130
2131 020256 012703 002700          BGNSEG  MOV  #CSPAT,R3          ;GET BEGINNING OF PATTERNS
2132 020262          EMT  CSBSEG          ;****START OF SEGMENT****
2133 020262 104004          1$:  MOV  #1776,R3          ;SET ALL SETTABLE BITS
2134 020264          MOV  #1776,GDDAT          ;SET UP EXPECT DATA IN
2135 020264 012777 001776 161634          BIC  (R3),GDDAT          ;GDDAT
2136 020272 012737 001776 002234          BIC  (R3),R3          ;CLEAR BITS IN RLCS VIA "BIC"
2137 020300 041337 002234          BIT  #DERR,R3          ;IF DRIVE ERROR BIT SET
2138 020304 041377 161616          BEQ  99$          ;EXPECT IT SET WHEN WE
2139 020310 032777 040000 161610          BIS  #ERR!DERR,GDDAT          ;READ IT BACK
2140 020316 001403          MOV  R3,R3          ;MOVE RLCS TO BDDAT FOR COMPARE
2141 020320 052737 140000 002234          BIC  #DRDY,BDDAT          ;CLEAR DRIVE READY
2142 020326 017737 161574 002236          CMP  BDDAT,GDDAT          ;DID "BIC" WORK PROPERLY
2143 020334 042737 000001 002236          BEQ  2$          ;BRANCH IF OKAY
2144 020342 023737 002236 002234
2145 020350 001404          ERDF  8,EM62,ERR2          ;WRONG DATA IN RLCS
2146          TRAP  T$ERRCODE
2147 020352 104462          .WORD  8
2148 020354 000010          .WORD  EM62
2149 020356 010132          .WORD  ERR2
2150 020360 011402          2$:  ESCAPE SEG          ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
2151 020362          EMT  C$ESCAPE
2152 020362 104010          .WORD  10000$-
2153 020364 000012
2154
2155
2156 020366 005723          TST  (R3)+          ;GET NEXT PATTERN
2157 020370 020327 002776          CMP  R3,#CSEND          ;AT END OF LIST
2158 020374 001333          BNE  1$          ;NO, GO BACK WITH NEXT PATTERN
2159 020376          ENOSEG 10000$:          ;****END OF SEGMENT****
2160 020376          EMT  C$ESEG
2161 020376 104005          ;****END OF TEST****
2162 020400          ENDTST
2163 020400          L10040: EMT  C$ETST
2164 020400 104001
2165
2166
2167 .SBTTL **TEST 17** - BIS OF RLBA
2168
2169 020402          BGNTST                      ;****START OF TEST****
2170
2171 020402          STARS
2172          ;*****
2173          ;TEST THAT THE "BIS" INSTRUCTION WILL WORK ON THE BUS
2174          ;ADDRESS REGISTER.  BITS 15-0 ARE LOADED, ONLY BITS 15-1

```

M05

PAGE: 0064AS
SEQ 0064

CVRLAA.P11 14-APR-78 15:04

TEST 17 - BIS OF RLBA

;ARE EXPECTED BACK. FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
;GROWING 0, AND SHIFTING 0.

STARS

;*****

2175									
2176									
2177	020402								
2178									
2179									
2180									
2181	020402	012703	002304						
2182	020406			BGNSEG	MOV	#BEGPAT,R3		;GET START OF LIST	
2183	020406	104004			EMT	CSBSEG		;*****START OF SEGMENT****	
2184	020410			1\$:					
2185	020410	005077	161514		CLR	RLBA		;CLEAR "BA"	
2186	020414	011337	002234		MOV	(R3),GDDAT		;SET EXPECTED	
2187	020420	005737	002260		TST	T,CNTLR		;RL11	
2188	020424	001403			BEQ	3\$		;NO	
2189	020426	042737	000001	002234	BIC	#1,GDDAT		;BIT 0 CAN'T SET IN RLBA (UNIBUS)	
2190	020434	051377	161470		BIS	(R3),RLBA		;BIS RLBA WITH PATTERN	
2191	020440	017737	161464	002236	MOV	RLBA,BDDAT		;READ "BA"	
2192	020446	023737	002236	002234	CMP	BDDAT,GDDAT		;DID RLBA LOAD PROPERLY?	
2193	020454	001404			BEQ	2\$		;BRANCH IF YES	
2194									
2195	020456				ERRDF	9,EM63,ERR2		;WRONG DATA IN RLBA	
2196	020456	104462			TRAP	T\$ERRCODE			
2197	020460	000011			.WORD	9			
2198	020462	010215			.WORD	EM63			
2199	020464	011402			.WORD	ERR2			
2200	020466			2\$:	ESCAPE	SEG		;IF /FL:LOE SET LOOP, ELSE EXIT SEG	
2201	020466	104010			EMT	C\$ESCAPE			
2202	020470	000012			.WORD	10000\$-			
2203									
2204	020472	005723			TST	(R3)+		;GET NEXT PATTERN	
2205	020474	020327	002512		CMP	R3,#ENDPAT		;DID WE COMPLETE LIST	
2206	020500	001343			BNE	1\$		;NO, GO BACK FOR NEXT.	
2207	020502			ENDSEG				;*****END OF SEGMENT****	
2208	020502			10000\$:					
2209	020502	104005			EMT	C\$ESEG			
2210	020504			ENDTST				;*****END OF TEST****	
2211	020504			L10041:					
2212	020504	104001			EMT	C\$ETST			
2213									
2214									
2215				.SBTTL	**TEST 18** - BIC OF RLBA				
2216									
2217	020506			BGNTST				;*****START OF TEST****	
2218									
2219	020506			STARS					
2220									
2221									
2222									
2223									
2224	020506								
2225									
2226									
2227									
2228	020506	012703	002304						
2229	020512			BGNSEG	MOV	#BEGPAT,R3		;GET START OF LIST	
2230	020512	104004			EMT	CSBSEG		;*****START OF SEGMENT****	

CVRLAA.P11 14-APR-78 15:04

TEST 18 - BIC OF RLBA

```

2231 020514 012777 177776 161406 1S: MOV #2,RLBA ;SET RLBA TO ALL 1'S (BIT 0=0)
2232 020514 012737 177776 002234 MOV #2,GDDAT ;SET UP EXPECTED RESULTS
2233 020522 041337 002234 BIC (R3),GDDAT ;IN GDDAT
2234 020530 041377 161370 BIC (R3),RLBA ;BIC RLBA
2235 020534 017737 161364 002236 MOV RLBA,BDDAT ;READ RLBA
2236 020540 023737 002236 002234 CMP BDDAT,GDDAT ;BIC WORK OKAY?
2237 020546 001404 BEQ 2S ;IF YES BRANCH
2238 020554 104462 ERRDF 10,EM64,ERR2 ;WRONG DATA IN RLBA
2239 020556 000012 TRAP TSEARCHCODE
2240 020560 010276 .WORD 10
2241 020562 011402 .WORD EM64
2242 020564 .WORD ERR2
2243 020566 104010 2S: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
2244 020570 000012 EMT CSESCAPE
2245 020572 005723 .WORD 10000$-
2246 020574 020327 002512 TST (R3)+ ;GET NEXT PATTERN
2247 020600 001345 CMP R3,#ENDPAT ;HAVE WE COMPLETED LIST
2248 020602 104005 BNE 1S ;NO, GO BACK FOR NEXT
2249 020604 104001 ENDSEG ;****END OF SEGMENT****
2250 020606 104001 10000$: EMT CSESEG ;****END OF TEST****
2251 020608 104001 ENDTST
2252 020610 104001 L10042: EMT CSETST
2253 020612 104001 .SBTTL **TEST 19** - BIS OF RLDA
2254 020614 104001 BGNSTST ;****START OF TEST****
2255 020616 104001 STARS
2256 020618 104001 ;*****
2257 020620 104001 ;TEST THAT THE "BIS" INSTRUCTION WILL WORK ON THE DISK ADDRESS
2258 020622 104001 ;REGISTER. BITS 15-0 ARE TESTED WITH 4 PATTERNS, GROWING 1,
2259 020624 104001 ;SHIFTING 1, GROWING 0, AND SHIFTING 0.
2260 020626 104001 STARS
2261 020628 104001 ;*****
2262 020630 104001 BGNSEG MOV #BEGPAT,R3 ;GET START OF LIST
2263 020632 104004 EMT CSBSEG ;****START OF SEGMENT****
2264 020634 104004 1S: CLR RLDA ;CLEAR "DA"
2265 020636 005077 161312 MOV (R3),GDDAT ;SET EXPECTED
2266 020638 011337 002234 BIS (R3),RLDA ;BIS RLDA
2267 020640 051377 161302 MOV RLDA,BDDAT ;READ RLDA
2268 020642 017737 161276 002236 CMP BDDAT,GDDAT ;IS RLDA CORRECT
2269 020644 023737 002236 002234 BEQ 2S ;IF OKAY BRANCH
2270 020646 001404 ERRDF 11,EM65,ERR2 ;WRONG DATA IN RLDA
2271 020648 104462 TRAP TSEARCHCODE
2272 020650 000013 .WORD 11

```

CVRLAA.P11 14-APR-78 15:04

TEST 19 - BIS OF RLDA

2287 020652 010361
 2288 020654 011402
 2289 020656 104010
 2290 020656 000012
 2291 020660 000012
 2292
 2293 020662 005723
 2294 020664 020327 002512
 2295 020670 001351
 2296 020672
 2297 020672 104005
 2298 020672
 2299 020674
 2300 020674 104001
 2301 020674
 2302
 2303
 2304
 2305
 2306 020676
 2307
 2308 020676
 2309
 2310
 2311
 2312
 2313 020676
 2314
 2315
 2316
 2317 020676 012703 002304
 2318 020702
 2319 020702 104004
 2320 020704
 2321 020704 012777 177777 161220
 2322 020712 012737 177777 002234
 2323 020720 041337 002234
 2324 020724 041377 161202
 2325 020730 017737 161176 002236
 2326 020736 023737 002234 002236
 2327 020744 001404
 2328
 2329 020746
 2330 020746 104462
 2331 020750 000014
 2332 020752 010442
 2333 020754 011402
 2334 020756 104010
 2335 020756 000012
 2336 020760
 2337
 2338 020762 005723
 2339 020764 020327 002512
 2340 020770 001345
 2341 020772
 2342 020772

2\$: .WORD EM65
 .WORD ERR2
 ESCAPE SEG ; IF /FL:LOE SET LOOP, ELSE EXIT SEG
 EMT C\$ESCAPE
 .WORD 10000\$-.
 TST (R3)+ ; GET NEXT PATTERN
 CMP R3, #ENDPAT ; HAVE WE FINISHED?
 BNE 1\$; NO GO BACK
 ENDSEG ; *****END OF SEGMENT*****
 10000\$: EMT C\$ESEG
 ENDTST ; *****END OF TEST*****
 L10043: EMT C\$ETST

.SBTTL **TEST 20** - BIC OF RLDA

BGNST ; *****START OF TEST*****

STARS

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

BGNSEG MOV #BEGPAT, R3 ; GET START OF LIST
 EMT C\$BSEG ; *****START OF SEGMENT*****
 1\$: MOV #-1, R1 ; SET RLDA TO ALL 1'S
 MOV #-1, GDOAT ; SET EXPECTED DATA
 BIC (R3), GDOAT ; SET EXPECTED DATA
 BIC (R3), R1 ; "BIC" RLDA
 MOV R1, BODAT ; READ RLDA
 CMP GDOAT, BODAT ; DID "BIC" WORK?
 BEQ 2\$; IF IT DID BRANCH
 ERDF 12, EM66, ERR2 ; WRONG DATA IN RLDA
 TRAP T\$ERRCODE
 .WORD 12
 .WORD EM66
 .WORD ERR2
 2\$: ESCAPE SEG ; IF /FL:LOE SET LOOP, ELSE EXIT SEG
 EMT C\$ESCAPE
 .WORD 10000\$-.
 TST (R3)+ ; GET NEXT PATTERN
 CMP R3, #ENDPAT ; DONE?
 BNE 1\$; NO GO BACK
 ENDSEG ; *****END OF SEGMENT*****
 10000\$:

CVRLAA.P11 14-APR-78 15:04

TEST 20 - BIC OF RLDA

```

EMT      CSESEG
ENDTST
L10044:   ;****END OF TEST****
EMT      CSETST

```

.SBTTL **TEST 21** - BUS RESET OF RLCS

```

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

```

STARS

```

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

```

STARS

;*****

```

SETPRI   #PRI07      ;PRIORITY TO SEVEN
MOV      #PRI07,RO
EMT      CSSPRI
MOV      #377,RLCS   ;LOAD ALL RLCS LOADABLE BITS
MOV      #CRDY,GDDAT ;SETUP EXPECTED
BIT      #DERR,RLCS  ;DRIVE ERR SET?
BEQ      1$          ;IF NOT DON'T EXPECT IT
BIS      #DERR!ERR,GDDAT ;IT'S SET, INIT BETTER NOT CLR
MOV      #100,RO     ;SET UP A WAIT LOOP
BRESET   ;BUS RESET
EMT      CSRESET
DEC      RO           ;WAIT IN CASE OF DRIVE ERROR
BNE      2$
MOV      #RLCS,BDDAT ;READ RLCS
BIC      #DRDY,BDDAT ;CLEAR OUT DRDY - DON'T CARE
CMP      BDDAT,GDDAT ;DID INIT WORK
BEQ      3$          ;YES, BRANCH
ERRDF    13,EM67,ERR2 ;WRONG DATA IN RLCS
TRAP     T$ERRCODE
.WORD    13
.WORD    EM67
.WORD    ERR2

```

3\$:

```

ENDTST
L10045:   ;****END OF TEST****

```

```

EMT      CSETST

```

.SBTTL **TEST 22** - BUS RESET OF RLBA

```

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

```

```

2343 020772 104005
2344 020774
2345 020774
2346 020774 104001
2347
2348
2349
2350
2351 020776
2352
2353 020776
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363 020776
2364
2365
2366
2367 020776
2368 020776 012700 000340
2369 021002 104041
2370 021004 012777 000377 161114
2371 021012 012737 000200 002234
2372 021020 032777 040000 161100
2373 021026 001403
2374 021030 052737 140000 002234
2375 021036 012700 000100
2376 021042
2377 021042 104033
2378 021044 005300
2379 021046 001376
2380 021050 017737 161052 002236
2381 021056 042737 000001 002236
2382 021064 023737 002236 002234
2383 021072 001404
2384
2385 021074
2386 021074 104462
2387 021076 000015
2388 021100 010525
2389 021102 011402
2390 021104
2391 021104
2392 021104
2393 021104 104001
2394
2395
2396
2397
2398 021106

```

CVRLAA.P11 14-APR-78 15:04

TEST 22 - BUS RESET OF RLBA

STARS

```

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

```

STARS

```

*****

```

021106 012777 177776 161014

021114 005737 002260

021120 001403

021122 052777 000001 161000

021130 005037 002234

021134

021134 104033

021136 017737 160766 002236

021144 001404

021146

021146 104462

021150 000016

021152 010562

021154 011402

021156

021156

021156

021156 104001

```

MOV      #-2, JRLBA      ;SET BA TO ALL 1'S
TST      T.CNTRL         ;RL11??
BEQ      2$              ;NO
BIS      #1, JRLBA
CLR      GDDAT            ;CLEAR EXPECTED DATA
BRESET   ;ISSUE BUS INIT
EMT      CSRESET
MOV      JRLBA, BDDAT     ;READ RLBA
BEQ      1$              ;IF CLEAR BRANCH

ERRDF    14, EM70, ERR2   ;WRONG DATA IN RLBA
TRAP     TSEARCHCODE
.WORD    14
.WORD    EM70
.WORD    ERR2

```

1\$:

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

L10046:

EMT CSETST

.SBTTL **TEST 23** - BUS RESET OF RLDA

BGNTST

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

STARS

```

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

```

STARS

```

*****

```

021160 012777 177777 160744

021166 005037 002234

021172

021172 104033

021174 017737 160732 002236

021202 001404

021204

021204 104462

021206 000017

021210 010617

```

MOV      #-1, JRLDA      ;SET DA TO ALL 1'S
CLR      GDDAT            ;CLEAR EXPECTED
BRESET   ;ISSUE BUS INIT
EMT      CSRESET
MOV      JRLDA, BDDAT     ;READ RLDA
BEQ      1$              ;IF CLEAR BRANCH

ERRDF    15, EM71, ERR2   ;WRONG DATA IN RLDA
TRAP     TSEARCHCODE
.WORD    15
.WORD    EM71

```


CVRLAA.P11 14-APR-78 15:04

TEST 23 - BUS RESET OF RLDA

1S: .WORD ERR2

ENDTST ;****END OF TEST****
L10047:

EMT CSETST

.SBTTL **TEST 24** - 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 REMAINED UNEFFECTED

```

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

```

```

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

```

ERRDF 16,EM72,ERR2 ;CS MODIFIED BA

TRAP T\$ERRCODE

.WORD 16

.WORD EM72

.WORD ERR2

1S: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT CSCLP1

```

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

```

```

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

```

ERRDF 17,EM73,ERR2 ;CS MODIFIED DA

TRAP T\$ERRCODE

.WORD 17

.WORD EM73

.WORD ERR2

2S:

2455 021212 011402

2456 021214

2457 021214

2458 021214

2459 021214

2460 021214 104001

2461 021214

2462 021214

2463 021214

2464 021214

2465 021216

2466 021216

2467 021216

2468 021216

2469 021216

2470 021216

2471 021216

2472 021216

2473 021216

2474 021216

2475 021216

2476 021216

2477 021216

2478 021216

2479 021224

2480 021232

2481 021240

2482 021240

2483 021240

2484 021240

2485 021246

2486 021254

2487 021254

2488 021256

2489 021264

2490 021264

2491 021272

2492 021272

2493 021274

2494 021276

2495 021300

2496 021302

2497 021302

2498 021302

2499 021304

2500 021312

2501 021312

2502 021314

2503 021322

2504 021322

2505 021330

2506 021330

2507 021332

2508 021334

2509 021336

2510 021340

CVRLAA.P11 14-APR-78 15:04

TEST 24 - UNIQUENESS OF RLCS

```

2511
2512
2513 021340
2514 021340
2515 021340 104001
2516
2517
2518
2519
2520
2521 021342
2522 021342
2523
2524
2525
2526
2527
2528 021342
2529
2530
2531
2532 021342 012737 000200 002234
2533 021350 032777 040000 160550
2534 021356 001403
2535 021360 052737 140000 002234
2536 021366 013777 002234 160532
2537 021374 012777 177777 160530
2538 021402 005077 160522
2539
2540
2541
2542
2543 021406 017737 160514 002236
2544 021414 042737 000001 002236
2545 021422 023737 002236 002234
2546 021430 001404
2547 021432
2548 021432 104462
2549 021434 000022
2550 021436 010742
2551 021440 011402
2552 021442
2553 021442 104006
2554
2555 021444 022777 177777 160460
2556
2557 021452 001412
2558
2559 021454 012737 177777 002234
2560 021462 017737 160444 002236
2561
2562 021470
2563 021470 104462
2564 021472 000023
2565 021474 010774
2566 021476 011402

```

```

ENDTST
L10050:
EMT CSETST

```

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

.SBTTL **TEST 25** - UNIQUENESS OF RLBA

```

BGNST
STARS

```

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

```

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

```

```

MOV #CRDY,GDDAT ;CONTROLLER READY
BIT #DERR,RLCS ;IF DRIVE ERROR IS
BEQ 99$ ;SET THEN EXPECT IT
BIS #ERR!DERR,GDDAT ;SET WHEN WE READ IT.
99$: MOV GDDAT,RLCS ;LOAD RLCS
MOV #-1,RLDA ;LOAD RLDA
CLR RLBA ;CLEAR RLBA

```

;CHECK IF RLCS IS OKAY

```

MOV RLCS,BDDAT ;READ RLCS
BIC #DRDY,BDDAT ;IGNORE DRIVE READY
CMP BDDAT,GDDAT ;CS OK?
BEQ 1$ ;YES, GO CHECK DA

```

;BA MODIFIED CS

```

ERRDF 18,EM74,ERR2
TRAP T$ERRCODE
.WORD 18
.WORD EM74
.WORD ERR2
1$: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT CSCLP1

```

;IS RLDA OKAY?

;IF OKAY BRANCH

```

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

```

;BA MODIFIED DA

```

ERRDF 19,EM75,ERR2
TRAP T$ERRCODE
.WORD 19
.WORD EM75
.WORD ERR2

```


CVRLAA.P11 14-APR-78 15:04

TEST 25 - UNIQUENESS OF RLBA

```

25:
ENDTST
L10051:
          EMT      CSETST
          ;*****END OF TEST*****

```

.SBTTL **TEST 26** - UNIQUENESS OF RLDA

```

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

```

STARS

```

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

```

2567	021500								
2568	021500								
2569	021500								
2570	021500	104001							
2571									
2572									
2573									
2574									
2575	021502								
2576									
2577									
2578	021502								
2579									
2580									
2581									
2582									
2583									
2584									
2585	021502								
2586									
2587									
2588									
2589	021502	012737	000200	002234		MOV	%CRDY,%DDAT		;CONTROLLER READY
2590	021510	032777	040000	160410		BIT	%DERR,%RLCS		;IF DRIVE ERROR SET
2591	021516	001403				BEQ	99S		;THEN EXPECT IT LATER
2592	021520	052737	140000	002234		BIS	%ERR,%DERR,%DDAT		
2593	021526	013777	002234	160372	99S:	MOV	%DDAT,%RLCS		;LOAD CS
2594	021534	012777	177776	160366		MOV	%-2,%RLBA		;LOAD BA WITH ALL 1'S
2595	021542	005077	160364			CLR	%RLDA		;CLEAR RLDA
2596									
2597									
2598									
2599	021546	017737	160354	002236		MOV	%RLCS,%DDAT		;READ RLCS
2600	021554	042737	000001	002236		BIC	%DRDY,%DDAT		;IGNORE DRIVE READY
2601	021562	023737	002234	002236		CMP	%DDAT,%DDAT		;RLCS OKAY?
2602	021570	001404				BEQ	1S		;YES, THEN BRANCH
2603									
2604	021572					ERRDF	20,%EM76,%ERR2		;DA MODIFIED CS
2605	021572	104462				TRAP	T\$ERRCODE		
2606	021574	000024				.WORD	20		
2607	021576	011026				.WORD	EM76		
2608	021600	011402				.WORD	ERR2		
2609	021602				1S:	CKLOOP			;CHECK IF /FL:LOE IS SET
2610	021602	104006				EMT	C\$CLP1		
2611									
2612	021604	022777	177776	160316		CMP	%-2,%RLBA		;IS RLBA OKAY?
2613	021612	001412				BEQ	2S		;BRANCH IF OKAY
2614									
2615	021614	012737	177776	002234		MOV	%-2,%DDAT		;SET UP EXPECTED
2616	021622	017737	160302	002236		MOV	%RLBA,%DDAT		;READ RLBA
2617									
2618	021630					ERRDF	21,%EM77,%ERR2		;DA MODIFIED BA
2619	021630	104462				TRAP	T\$ERRCODE		
2620	021632	000025				.WORD	21		
2621	021634	011061				.WORD	EM77		
2622	021636	011402				.WORD	ERR2		

CVRLAA.P11 14-APR-78 15:04

TEST 26 - UNIQUENESS OF RLDA

25:

ENDTST ;****END OF TEST****
L10052:

EMT CSETST

.SBTTL **TEST 27** - UNIQUENESS OF RLMP

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

STARS

```

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

```

```

2623 021640
2624
2625
2626 021640
2627 021640
2628 021640 104001
2629
2630
2631
2632 021642
2633
2634
2635 021642
2636
2637
2638
2639
2640
2641 021642
2642
2643
2644
2645 021642 012737 000200 002234
2646 021650 032777 040000 160250
2647 021656 001403
2648 021660 052737 140000 002234
2649 021666 013777 002234 160232
2650 021674 012777 177776 160226
2651 021702 012777 177777 160222
2652 021710 005077 160220
2653
2654
2655
2656 021714 017737 160206 002236
2657 021722 042737 000001 002236
2658 021730 023737 002234 002236
2659 021736 001404
2660
2661 021740
2662 021740 104462
2663 021742 000026
2664 021744 007730
2665 021746 011402
2666 021750
2667 021750 104006
2668
2669 021752 022777 177776 160150
2670 021760 001412
2671
2672 021762 012737 177776 002234
2673 021770 017737 160134 002236
2674
2675 021776
2676 021776 104462
2677 022000 000027
2678 022002 007763

```

```

MOV #CRDY,GDDAT ;CONTROLLER READY
BIT #DERR,RLCS ;IF DRIVE ERROR SET
BEQ 99$ ;THE EXPECT IT LATER
BIS #ERR!DERR,GDDAT
99$: MOV GDDAT,RLCS ;LOAD CS
MOV #-2,RLBA ;LOAD BA WITH ALL 1'S
MOV #-1,RLDA ;LOAD RLDA
CLR RLMP ;WRITE RLMP

```

;CHECK IF RLCS IS OKAY

```

MOV RLCS,BDDAT ;READ RLCS
BIC #DRDY,BDDAT ;IGNORE DRIVE READY
CMP GDDAT,BDDAT ;RLCS OKAY?
BEQ 1$ ;YES, THEN BRANCH

ERRDF 22,EM44,ERR2 ;MP MODIFIED CS
TRAP T$ERRCODE
.WORD 22
.WORD EM44
.WORD ERR2
1$: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT CSCLP1

CMP #-2,RLBA ;IS RLBA OKAY?
BEQ 2$ ;BRANCH IF OKAY

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

ERRDF 23,EM45,ERR2 ;MP MODIFIED BA
TRAP T$ERRCODE
.WORD 23
.WORD EM45

```


CVRLAA.P11 14-APR-78 15:04

TEST 27 - UNIQUENESS OF RLMP

```

2679 022004 011402
2680 022006
2681 022006 104006
2682 022010 022777 177777 160114
2683 022016 001412
2684
2685 022020 017737 160106 002236
2686 022026 012737 177777 002234
2687
2688 022034
2689 022034 104462
2690 022036 000030
2691 022040 010016
2692 022042 011402
2693
2694 022044
2695
2696
2697 022044
2698 022044
2699 022044 104001
2700
2701
2702
2703 022046
2704
2705
2706
2707 022046
2708
2709
2710
2711
2712
2713 022046
2714
2715
2716
2717 022046 005737 002260
2718 022052 001410
2719
2720
2721 022054 004537 014602
2722 022060 000000
2723 022062 004537 016354
2724 022066
2725 022066 104006
2726
2727 022070 004537 014302
2728
2729 022074
2730 022074
2731 022074
2732 022074 104001
2733
2734

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

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

        ERRDF    24, EM46, ERR2
        TRAP     T$EACODE
        .WORD    24
        .WORD    EM46
        .WORD    ERR2

3$:

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

.SBTTL  **TEST 28** - 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
        BEQ      99$
                ;RLV11??
                ;YES SKIP TEST

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

        JSR      R5, CHERR
                ;CHECK CONTROLLER FOR ERRORS

99$:
ENDTST
L10054:
        EMT      C$ETST
                ;*****END OF TEST****

```

CVRLAA.P11 14-APR-78 15:04

TEST 29 - TEST NOOP DOES NOTHING (RL11 ONLY)

.SBTTL **TEST 29** - TEST NOOP DOES NOTHING (RL11 ONLY)

BGNTST

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

STARS

```

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

```

```

2735
2736
2737 022076
2738
2739 022076
2740
2741
2742
2743 022076
2744
2745
2746 022076 005737 002260
2747 022102 001476
2748
2749 022104 012777 000001 160020
2750 022112 012777 000002 160010
2751 022120 005077 160010
2752 022124 017737 160004 002234
2753
2754 022132 004537 014602
2755 022136 000000
2756 022140 004537 016354
2757 022144
2758 022144 104006
2759
2760 022146 004537 014302
2761 022152
2762 022152 104010
2763 022154 000124
2764
2765 022156 017737 157752 002236
2766 022164 023737 002234 002236
2767 022172 001404
2768
2769 022174
2770 022174 104462
2771 022176 000031
2772 022200 006763
2773 022202 011402
2774
2775 022204
2776 022204 104006
2777
2778 022206 012737 000002 002234
2779 022214 017737 157710 002236
2780 022222 023737 002234 002236
2781 022230 001404
2782
2783 022232
2784 022232 104462
2785 022234 000032
2786 022236 007011
2787 022240 011402
2788
2789 022242
2790 022242 104006

```

```

TST T.CNTRL ;RLV11??
BEQ 3$

MOV #1, JRLDA ;LOAD DISK ADDRESS
MOV #2, JRLBA ;LOAD BUS ADDRESS
CLR JRLMP
MOV JRLMP, GDDAT ;READ RLMP

JSR R5, LDFUNC ;ISSUE FUNCTION OF FOLLOWING WORD
NOOP0
JSR R5, WTCRDY ;WAIT FOR CONTROLLER READY HIGH
CKLOOP ;CHECK IF /FL:LOE IS SET
EMT CSCLP1

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

MOV JRLMP, BDDAT ;READ RLMP
CMP GDDAT, BDDAT ;RLMP OK?
BEQ 1$

ERRDF 25, EM14, ERR2
TRAP T$ERRCODE
.WORD 25
.WORD EM14
.WORD ERR2

1$: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT CSCLP1

MOV #2, GDDAT ;SET UP EXP'D BA
MOV JRLBA, BDDAT ;READ BA
CMP GDDAT, BDDAT ;BA OK?
BEQ 2$ ;YES

ERRDF 26, EM15, ERR2
TRAP T$ERRCODE
.WORD 26
.WORD EM15
.WORD ERR2

2$: CKLOOP ;CHECK IF /FL:LOE IS SET
EMT CSCLP1

```


CVRLAA.P11 14-APR-78 15:04

TEST 29 - TEST NOOP DOES NOTHING (RL11 ONLY)

```

2791
2792 022244 012737 000001 002234      MOV      #1,GDDAT      ;SET UP EXP'D DA
2793 022252 017737 157654 002236      MOV      JALDA,BDDAT    ;READ DA
2794 022260 023737 002234 002236      CMP      GDDAT,BDDAT    ;DA OKAY
2795 022266 001404                      BEQ      3$
2796
2797 022270                      ERRDF   27.,EM16,ERR2
2798 022270 104462                      TRAP   T$ERRCODE
2799 022272 000033                      .WORD  27
2800 022274 007037                      .WORD  EM16
2801 022276 011402                      .WORD  ERR2

```

3\$:

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

L10055:

EMT C\$ETST

.SBTTL **TEST 30** - TEST OF INTERRUPT (RL11 ONLY)

BGNTST

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

STARS

```

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

```

```

2829 022302 005737 002260      TST      T.CNTRL
2830 022306 001426                      BEQ      99$
2831
2832 022310 005037 002176      CLR      INTFLG      ;CLEAR INTERRUPT OCCRUANCE FLAG
2833 022314                      SETPRI   #PRI00      ;SET PSW TO 0
2834 022314 012700 000000      MOV      #PRI00,RO
2835 022320 104041                      EMT      C$SPRI
2836 022322 004537 014602      JSR      R5,LDFUNC      ;ISSUE FUNCTION OF FOLLOWING WORD
2837 022326 000100                      NOOPD:INTEN      ;NOOP AND INTERRUPT ENABLE
2838 022330 004537 016354      JSR      R5,WTCDY      ;WAIT FOR CONTROLLER READY HIGH
2839 022334 005737 002176      TST      INTFLG      ;DID INTERRUPT OCCUR
2840 022340 001004                      BNE     2$          ;IF SO BRANCH
2841 022342                      ERRDF   28.,EM13,ERR0
2842 022342 104462                      TRAP   T$ERRCODE
2843 022344 000034                      .WORD  28
2844 022346 006731                      .WORD  EM13
2845 022350 011352                      .WORD  ERR0
2846 022352 005037 002176      2$:    CLR      INTFLG

```

CVRLAA.P11 14-APR-78 15:04

TEST 30 - TEST OF INTERRUPT (RL11 ONLY)

2847	022356			CKLOOP		;CHECK IF /FL:LOE IS SET
2848	022356	104006		EMT	C\$CLP1	
2849	022360	004537	014302	JSR	RS,CHERR	;CHECK CONTROLLER FOR ERRORS

2850						
2851						
2852	022364			99\$:		
2853	022364			ENDTST		;****END OF TEST****
2854	022364			L10056:		
2855	022364	104001		EMT	CSETST	
2856						
2857						

.SBTTL **TEST 31** - TEST PRIORITY BR LEVEL (RL11 ONLY)

2858						
2859						
2860	022366			BGNTST		;****START OF TEST****
2861						
2862	022366					

STARS

```

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

```

STARS

```

;*****

```

2869						
2870	022366	005737	002260	TST	T.CNTR	;RLV11??
2871	022372	001456		BEG	6\$	;YES, SKIP TEST
2872						
2873	022374	012737	000340	MOV	#340,BDDAT	;SET UP INITIAL OF 7
2874	022402	013737	002140	MOV	JPRIOR,GDDAT	;GET GIVEN PRIORITY
2875						
2876	022410			BGNSEG		;****START OF SEGMENT****
2877	022410	104004		EMT	C\$BSEG	
2878						
2879	022412	005037	002176	SS:	CLR	INTFLG
2880	022416			SETPRI	BDDAT	;CLEAR INTERRUPT OCCURANCE
2881	022416	013700	002236	MOV	BDDAT,RO	;SET PRIORITY
2882	022422	104041		EMT	C\$SPR1	
2883						
2884	022424	004537	014602	JSR	RS,LDFUNC	;ISSUE FUNCTION OF FOLLOWING WORD
2885	022430	000100		NOOPD:INTEN		
2886						
2887	022432	004537	016354	JSR	RS,WTCRDY	;WAIT FOR CONTROLLER READY HIGH
2888	022436			ESCAPE	TST	;IF /FL:LOE SET LOOP, ELSE EXIT TST
2889	022436	104010		EMT	C\$ESCAPE	
2890	022440	000070		.WORD	L10057-	
2891						
2892	022442	004537	014302	JSR	RS,CHERR	;CHECK CONTROLLER FOR ERRORS
2893	022446			ESCAPE	TST	;IF /FL:LOE SET LOOP, ELSE EXIT TST
2894	022446	104010		EMT	C\$ESCAPE	
2895	022450	000060		.WORD	L10057-	
2896						
2897	022452	023737	002236	CMP	BDDAT,GDDAT	;SHOULD IT INTERRUPT
2898	022460	002012	002234	BGE	1\$	;NO, BRANCH
2899						
2900	022462	005737	002176	TST	INTFLG	;DID INTERRUPT OCCUR
2901	022466	001004		BNE	2\$	;YES, OK
2902						

CVRLAA.P11 14-APR-78 15:04

TEST 31 - TEST PRIORITY BR LEVEL (RL11 ONLY)

2903	022470			3\$:	ERRDF	29	EM17,ERR7	
2904	022470	104462			TRAP		T\$ERCODE	
2905	022472	000035			.WORD	29		
2906	022474	007065			.WORD	EM17		
2907	022476	011670			.WORD	ERR7		
2908				2\$:	ESCAPE	SEG		;IF /FL:LOE SET LOOP, ELSE EXIT SEG
2909	022500				EMT	C\$ESCAPE		
2910	022500	104010			.WORD	10000\$-		
2911	022502	000014			BR	4\$		
2912	022504	000405		1\$:	TST	INTFLG		;DID INTERRUPT OCCUR
2913	022506	005737	002176		BEG	2\$		;NO, OK
2914	022512	001772			BR	3\$		;YES, ERROR
2915	022514	000765						
2916								
2917	022516			ENDSEG				;****END OF SEGMENT****
2918	022516			10000\$:				
2919	022516	104005			EMT	C\$ESEG		
2920	022520	162737	000040	4\$:	SUB	840,BDDAT		;NEXT LEVEL
2921	022526	100331			BPL	5\$		
2922								
2923	022530			6\$:				
2924	022530			ENDTST				;****END OF TEST****
2925	022530			L10057:				
2926	022530	104001			EMT	C\$ETST		
2927								
2928								
2929								
2930				.SBTTL	**TEST 32** - RLV11 MAINT. FORCED OPI TEST, LESS THAN 510 WORDS			
2931	022532			BGNTST				;****START OF TEST****
2932								
2933	022532			STARS				
2934				;;*****				
2935				;;PERFORM RLV11 MAINTENANCE FUNCTION 0 WITH LESS THAN 510 WORDS				
2936				;;TO FORCE OPI ERROR. THE TEST SHOULD FORCE COMPOSITE ERROR(BIT 15),				
2937				;;HEADER NOT FOUND(BIT12) AND OPI(BIT 10). DRIVE ERROR IS IGNORED.				
2938	022532			STARS				
2939				;;*****				
2940	022532	005737	002260		TST	T.CNTRL		;RLV11?
2941	022536	001040			BNE	10\$		;NO, EXIT TEST
2942	022540	012703	002514	1\$:	MOV	#PATCRC,R3		;GET CRC PATTERN TABLE
2943	022544	012704	002606		MOV	#PATDAT,R4		;GET DATA PATTERN TABLE
2944	022550	011337	022564		MOV	(R3),2\$		;STORE CRC PATTERN
2945	022554	011437	022572		MOV	(R4),3\$		;STORE DATA PATTERN
2946	022560	004537	015216		JSR	R5,CALCRC		;CALCULATE CRC BEFORE TEST
2947	022564	000000		2\$:	.WORD	0		
2948	022566	004537	015454		JSR	R5,SETPAT		;SETUP PATTERN BEFORE TEST
2949	022572	000000		3\$:	.WORD	0		
2950	022574				BGNSEG			
2951	022574	104004			EMT	C\$BSEG		
2952	022576	004537	015542		JSR	R5,LDFUN		;PERFORM MAINT FUNCTION
2953	022602	000000			MAINT			
2954	022604	177271			-507			;LESS THAN 510 WORDS
2955	022606	006044			MATMES			;MAINT. MESSAGE
2956	022610	004537	016354		JSR	R5,WTCRDY		
2957	022614				CKLOOP			;LOOP SWITCH
2958	022614	104006			EMT	C\$CLP1		

CVRLAA.P11 14-APR-78 15:04

TEST 32 - RLV11 MAINT. FORCED OPI TEST, LESS THAN 510 WORDS

```

2959 022616 004537 015026
2960 022622 000404
2961 022624
2962 022624 104462
2963 022626 000036
2964 022630 007557
2965 022632 011726
2966 022634
2967 022634 104006
2968 022636
2969 022636
2970 022636 104005
2971 022640
2972
2973 022640
2974 022640
2975 022640 104001
2976
2977
2978
2979 022642
2980
2981 022642
2982
2983
2984
2985
2986 022642
2987
2988 022642 005737 002260
2989 022646 001040
2990 022650 012703 002514
2991 022654 012704 002606
2992 022660 011337 022674
2993 022664 011437 022702
2994 022670 004537 015216
2995 022674 000000
2996 022676 004537 015454
2997 022702 000000
2998 022704
2999 022704 104004
3000 022706 004537 015542
3001 022712 000000
3002 022714 177266
3003 022716 006044
3004 022720 004537 016354
3005 022724
3006 022724 104006
3007 022726 004537 015026
3008 022732 000404
3009 022734
3010 022734 104462
3011 022736 000037
3012 022740 007633
3013 022742 011726
3014 022744

```

```

JSR R5,CHKOPI ;CHECK FOR EXPECTED ERRORS
BR 4$ ;EXPECTED ERRORS FOUND,EXIT TEST
ERRDF 30,EM27,ERR10
TRAP T$ERCODE
.WORD 30
.WORD EM27
.WORD ERR10
4$: CKLOOP
EMT C$CLP1
ENDSEG
10000$: EMT C$ESEG
10$:
ENDTST
L10060: EMT C$ETST

```

.SBTTL **TEST 33** - RLV11 MAINT. FORCED OPI TEST, MORE THAN 511 WORDS

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

STARS

```

;*****
;PERFORM RLV11 MAINTENANCE FUNCTION 0 WITH MORE THAN 511 WORDS
;TO FORCE OPI ERROR. THE TEST SHOULD FORCE COMPOSITE ERROR(BIT 15),
;HEADER NOT FOUND(BIT12) AND OPI(BIT 10). DRIVE ERROR IS IGNORED.
STARS
;*****

```

```

TST T,CNTRL ;RLV11?
BNE 10$ ;NO EXIT TEST
1$: MOV #PATCRC,R3 ;GET CRC PATTERN TABLE
MOV #PATDAT,R4 ;GET DATA PATTERN TABLE
MOV (R3),2$ ;STORE CRC PATTERN
MOV (R4),3$ ;STORE DATA PATTERN
JSR R5,CALCRC ;CALCULATE CRC BEFORE TEST
2$: .WORD 0
JSR R5,SETPAT ;SETUP PATTERN BEFORE TEST
3$: .WORD 0
BGNSEG
EMT C$BSEG
JSR R5,LDFUN ;PERFORM MAINT FUNCTION
MAINT -512 ;MORE THAN 511 WORDS
MATMES ;MAINT. MESSAGE
JSR R5,WTCRDY
CKLOOP ;LOOP SWITCH
EMT C$CLP1
JSR R5,CHKOPI ;CHECK FOR EXPECTED ERRORS
BR 4$ ;EXPECTED ERRORS FOUND,EXIT TEST
ERRDF 31,EM30,ERR10
TRAP T$ERCODE
.WORD 31
.WORD EM30
.WORD ERR10
4$: CKLOOP

```


CVRLAA.P11 14-APR-78 15:04

TEST 33 - RLV11 MAINT. FORCED OPI TEST, MORE THAN 511 WORDS

```

3015 022744 104006
3016 022746
3017 022746
3018 022746 104005
3019 022750
3020
3021 022750
3022 022750
3023 022750 104001
3024
3025
3026
3027 022752
3028
3029 022752
3030
3031
3032
3033
3034 022752
3035
3036
3037 022752 005737 002260
3038 022756 001052
3039 022760 012703 002514
3040 022764 012704 002606
3041 022770 011337 023004
3042 022774 011437 023012
3043 023000 004537 015216
3044 023004 000000
3045 023006 004537 015454
3046 023012 000000
3047 023014
3048 023014 104004
3049 023016
3050 023016 012700 000000
3051 023022 104041
3052 023024 005037 002176
3053 023030 004537 015542
3054 023034 000100
3055 023036 177266
3056 023040 006104
3057 023042 004537 016354
3058 023046
3059 023046 104006
3060 023050
3061 023050 012700 000340
3062 023054 104041
3063 023056 005737 002176
3064 023062 001004
3065 023064
3066 023064 104462
3067 023066 000040
3068 023070 007404
3069 023072 011352
3070 023074 005037 002176

```

```

      EMT      CSCLP1
      ENDSEG
10000$:
      EMT      CSESEG
10$:
      ENDTST
      L10061:
      EMT      CSETST
      .SBTTL   **TEST 34** - RLV11 MAINT. FORCED OPI TEST - INTERRUPT MODE
      BGNSTST                                     ;****START OF TEST****
      STARS
      ;*****
      ;PERFORM TEST OF INTERRUPT BY ISSUING RLV11 MAINTENANCE FUNCTION 0
      ;WITH MORE THAN 511 WORDS TO FORCE OPI ERROR. CHECK INTERRUPT
      ;OPERATION AND REPORT IF ERROR OCCURS.
      STARS
      ;*****
      TST      T.CNTRL      ;RLV11?
      BNE      10$         ;NO EXIT TEST
      1$:      MOV      #PATCRC,R3      ;GET CRC PATTERN TABLE
      MOV      #PATDAT,R4      ;GET DATA PATTERN TABLE
      MOV      (R3),2$        ;STORE CRC PATTERN
      MOV      (R4),3$        ;STORE DATA PATTERN
      JSR      RS,CALCRC      ;CALCULATE CRC
      2$:      .WORD      0
      JSR      RS,SETPAT      ;SETUP PATTERN
      3$:      .WORD      0
      BGNSEG
      EMT      CSBSEG
      SETPRI   #PRI00        ;SET PRIORITY TO ZERO
      MOV      #PRI00,R0
      EMT      CSSPRI
      CLR      INTFLG        ;CLEAR INT. FLAG
      JSR      RS,LDFUN      ;MAINT FUNCTION, INT DRIVEN
      MAINT!INTEN            ;MORE THAN 511 TO FORCE OPI ERROR
      -512
      MATINT
      JSR      RS,WTCRDY      ;WAIT FOR READY
      CKLOOP
      EMT      CSCLP1
      SETPRI   #PRI07
      MOV      #PRI07,R0
      EMT      CSSPRI
      TST      INTFLG        ;CHECK IF INTERRUPT OCCURRED
      BNE      4$
      ERROF    32,EM24,ERRO
      TRAP     T$ERCODE
      .WORD    32
      .WORD    EM24
      .WORD    ERRO
      4$:      CLR      INTFLG      ;CLEAR INT. FLAG

```

CVRLAA.P11 14-APR-78 15:04

TEST 34 - RLV11 MAINT. FORCED OPI TEST - INTERRUPT MODE

```

3071 023100
3072 023100 104006
3073 023102
3074 023102
3075 023102 104005
3076 023104
3077
3078 023104
3079 023104
3080 023104 104001
3081
3082
3083
3084
3085 023106
3086
3087 023106
3088
3089
3090
3091
3092 023106
3093
3094
3095 023106 005737 002260
3096 023112 001402
3097 023114 000137 023434
3098 023120 012703 002514
3099 023124 012704 002606
3100 023130 011337 023144
3101 023134 011437 023152
3102 023140 004537 015216
3103 023144 000000
3104 023146 004537 015454
3105 023152 000000
3106 023154
3107 023154 104004
3108 023156
3109 023156 013700 002142
3110 023162 104036
3111 023164
3112 023164 012746 000340
3113 023170 012746 016300
3114 023174 013746 002142
3115 023200 012746 000003
3116 023204 104037
3117 023206 062706 000010
3118 023212
3119 023212 012700 000000
3120 023216 104041
3121 023220 005037 002176
3122 023224 013700 002266
3123 023230 006300
3124 023232 006300
3125 023234 006300
3126 023236 063700 002266

```

```

CKLOOP
EMT CSCLP1
ENDSEG
10000S: EMT CSESEG
10S:
ENDTST
L10062: EMT CSETST

```

.SBTTL **TEST 35** - RLV11 OPI TIMEOUT TEST

BGNTST ;START OF TEST

STARS

```

;*****
;PERFORM RLV11 MAINTENANCE FUNCTION (0) WITH INTERRUPT MODE. FORCE
;OPI TIMEOUT BY SETTING WORD COUNT TO -512 WORDS. MEASURE OPI TIMEOUT
;AND COMPARE TO MIN. AND MAX. LIMITS.

```

STARS

;*****

```

1S: TST T.CNTRL ;RLV11?
    BEQ 1S ;YES, PERFORM TEST
    JMP 10S ;RLV11 EXIT TEST
    MOV #PATCRC,R3 ;GET CRC PATTERN TABLE
    MOV #PATDAT,R4 ;GET DATA PATTERN TABLE
    MOV (R3),2S ;STORE CRC PATTERN
    MOV (R4),3S ;STORE DATA PATTERN
    JSR RS,CALCRC ;CALCULATE CRC BEFORE TEST
2S: .WORD 0
    JSR RS,SETPAT ;SETUP PATTERN BEFORE TEST
3S: .WORD 0
    BGNSEG
    EMT CSBSEG
    CLAVEC BVEC ;CLEAR PRESENT INT. VECTOR
    MOV BVEC,R0
    EMT CSCVEC
    SETVEC BVEC,#TIMSRV,#340
    MOV #340,-(SP)
    MOV #TIMSRV,-(SP)
    MOV BVEC,-(SP)
    MOV #3,-(SP)
    EMT CSSVEC
    ADD #10,SP
    SETPRI #PRI00 ;SETUP FOR WAIT ABORT
    MOV #PRI00,R0
    EMT CSSPRI
    CLR INTFLG ;CLEAR INTERRUPT FLAG
    MOV OPIMX,R0 ;OPI LIMIT SETUP
    ASL R0
    ASL R0
    ASL R0
    ADD OPIMX,R0

```


CVRLAA.P11 14-APR-78 15:04

TEST 35 - RLV11 OPI TIMEOUT TEST

3127	023242	063700	002266		ADD	OPIMX,RO	
3128	023246	004537	015542		JSR	RS,LOFUN	
3129	023252	000100			MAINT!INTEN		;PERFORM MAINT. FUNCTION
3130	023254	177266			-512		;MAINT FUNCTION WITH INT.MODE
3131	023256	006104			MATINT		;WORD COUNT
3132	023260				WAITUS	RO	;MAINT MESSAGE
3133	023260	104027			EMT	CSWTU	;WAIT MAX. MILLISECONDS
3134	023262	005737	002176		TST	INTFLG	
3135	023266	001004			BNE	4S	;CHECK INT. FLG
3136	023270				ERRDF	33,EM24,ERR0	;ERROR ON INTERRUPT
3137	023270	104462			TRAP	TSEACODE	
3138	023272	000041			.WORD	33	
3139	023274	007404			.WORD	EM24	
3140	023276	011352			.WORD	ERR0	
3141	023300	005037	002176	4S:	CLR	INTFLG	
3142	023304				CKLOOP		
3143	023304	104006			EMT	CSCLP1	
3144	023306				GETTIM	BDDAT	;GET TIME EXPIRED
3145	023306	104052			EMT	CSGTIM	
3146	023310	010037	002236		MOV	RO,BDDAT	
3147	023314	005000			CLR	RO	;DIVIDE
3148	023316	162737	000012	002236	SUB	#10.,BDDAT	;ANSWER
3149	023324	100402			BMI	6S	;BY 10 TO GET
3150	023326	005200			INC	RO	;RIGHT ANSWER
3151	023330	000772			BR	5S	
3152	023332	010037	002236	6S:	MOV	RO,BDDAT	;STORE DIVIDED RESULT
3153							
3154							
3155							
3156	023336						
3157	023336	012700	000340	7S:	SETPRI	#PRI07	
3158	023342	104041			MOV	#PRI07,RO	
3159	023344	023737	002266	002236	EMT	CSSPRI	
3160	023352	002404			CMP	OPIMX,BDDAT	;IS OPI WITHIN LIMITS
3161	023354	023737	002264	002236	BLT	8S	;NO REPORT ERROR
3162	023362	003404			CMP	OPIMN,BDDAT	;WITHIN LIMITS?
3163	023364				BLE	9S	;YES
3164	023364	104462			ERRDF	34,EM31,ERR11	;OPI TIMING INCORRECT
3165	023366	000042			TRAP	TSEACODE	
3166	023370	007707			.WORD	34	
3167	023372	011770			.WORD	EM31	
3168	023374				.WORD	ERR11	
3169	023374	104006		9S:	CKLOOP		
3170	023376				EMT	CSCLP1	
3171	023376	013700	002142		CLRVEC	BVEC	;CLEAR PRESENT VECTOR AND RESET OLD
3172	023402	104036			MOV	BVEC,RO	
3173	023404				EMT	CSCVEC	
3174	023404	012746	000340		SETVEC	BVEC,#INTSRV,#340	
3175	023410	012746	016272		MOV	#340,-(SP)	
3176	023414	013746	002142		MOV	#INTSRV,-(SP)	
3177	023420	012746	000003		MOV	BVEC,-(SP)	
3178	023424	104037			MOV	#3,-(SP)	
3179	023426	062706	000010		EMT	CSSVEC	
3180	023432				ADD	#10,SP	
3181	023432			10000S:	ENDSEG		
3182	023432	104005			EMT	CSESEG	

CVRLAA.P11 14-APR-78 15:04

TEST 35 - RLV11 OPI TIMEOUT TEST

10\$:

ENDTST
L10063:

EMT CSETST

.SBTTL **TEST 36** - TEST RLV11 MAINT. FUNCTION - FLAG MODE

BGNTST

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

STARS

```

*****
:PERFORM RLV11 MAINTENANCE FUNCTION 0 IN FLAG MODE AND CHECK
:FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
:WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER
:THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
:RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
:FIFO INTO BUF2 MEMORY FOR PROPER DATA.
:CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
:A VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.

```

STARS

;*****

```

:RLV11?
:YES,RLV11
:NO,SKIP TEST
:GET CRC PATTERN TABLE
:GET DATA PATTERN TABLE
:STORE CRC PATTERN FOR CALCULATION
:STORE DATA PAT. FOR BUFFER FILL
:CALCULATE CRC BEFORE TEST
:PATTERN FOR CRC TEST

TST T.CNTRL
BEQ 100$
JMP 10$
100$: MOV #PATCRC,R3
      MOV #PATDAT,PATSAV
101$: MOV (R3),102$
      MOV #PATSAV,103$
      JSR R5,CALCRC
102$: .WORD 0
      BGNSEG
      EMT
      JSR R5,SETPAT
103$: .WORD 0
      JSR R5,LDFUN
      MAINT
      -511.
      MATMES
      JSR R5,WTCRDY
      CKLOOP
      EMT
      JSR R5,CHERR
      CKLOOP
      EMT
      CSCLP1
      MOV #BUF1+1776,GDDAT
      MOV E.BA,BDDAT
      CMP GDDAT,BDDAT
      BEQ 1$
      ERADF 35,EM10,ERR4
      TRAP T$ERRCODE
      .WORD 35
      .WORD EM10
      .WORD ERR4
1$: CKLOOP

```

;CHECK FOR LOOP MODE

```

3183 023434
3184
3185 023434
3186 023434
3187 023434 104001
3188
3189
3190
3191
3192 023436
3193
3194 023436
3195
3196
3197
3198
3199
3200
3201
3202
3203
3204 023436
3205
3206 023436 005737 002260
3207 023442 001402
3208 023444 000137 024246
3209 023450 012703 002514
3210 023454 012737 002606 002300
3211 023462 011337 023500
3212 023466 017737 156606 023510
3213 023474 004537 015216
3214 023500 000000
3215 023502
3216 023502 104004
3217 023504 004537 015454
3218 023510 000000
3219 023512 004537 015542
3220 023516 000000
3221 023520 177001
3222 023522 006044
3223 023524 004537 016354
3224 023530
3225 023530 104006
3226 023532 004537 014302
3227 023536
3228 023536 104006
3229 023540 012737 005476 002234
3230 023546 013737 002162 002236
3231 023554 023737 002234 002236
3232 023562 001404
3233 023564
3234 023564 104462
3235 023566 000043
3236 023570 006566
3237 023572 011546
3238 023574

```


CVRLAA.P11 14-APR-78 15:04

TEST 36 - TEST RLV11 MAINT. FUNCTION - FLAG MODE

3239	023574	104006			EMT	C\$CLP1	
3240	023576	013737	002152	002234	MOV	B.DA,GDDAT	;GET BEFORE DA REGISTER
3241	023604	013737	002164	002236	MOV	E.DA,BDDAT	
3242	023612	005037	002222		CLR	TEMP1	
3243	023616	113737	002152	002222	MOVB	B.DA,TEMP1	
3244	023624	062737	000006	002222	ADD	#6,TEMP1	;+6 TO DA LOW BYTE
3245	023632	113737	002222	002234	MOVB	TEMP1,GDDAT	;STORE LOW BYTE OF DA
3246	023640	023737	002234	002236	CMP	GDDAT,BDDAT	
3247	023646	001404			BEQ	2\$	
3248	023650				ERRDF	36.EM12,ERR4	
3249	023650	104462			TRAP	T\$ERRCODE	
3250	023652	000044			.WORD	36	
3251	023654	006670			.WORD	EM12	
3252	023656	011546			.WORD	ERR4	
3253	023660				CKLOOP		
3254	023660	104006			EMT	C\$CLP1	
3255	023662	013737	002242	002234	MOV	GDCRCA,GDDAT	;GET CRC OF DA+3 VALUE
3256	023670	013737	002166	002236	MOV	E.MP,BDDAT	;GET CONTROLLER CRC OF DA+3
3257	023676	023737	002234	002236	CMP	GDDAT,BDDAT	
3258	023704	001404			BEQ	3\$	
3259	023706				ERRDF	37.EM20,ERR4	
3260	023706	104462			TRAP	T\$ERRCODE	
3261	023710	000045			.WORD	37	
3262	023712	007120			.WORD	EM20	
3263	023714	011546			.WORD	ERR4	
3264	023716				CKLOOP		
3265	023716	104006			EMT	C\$CLP1	
3266	023720	013737	002244	002234	MOV	GDCRCB,GDDAT	
3267	023726	013737	002170	002236	MOV	E.MP1,BDDAT	
3268	023734	023737	002234	002236	CMP	GDDAT,BDDAT	
3269	023742	001404			BEQ	4\$	
3270	023744				ERRDF	38.EM21,ERR4	
3271	023744	104462			TRAP	T\$ERRCODE	
3272	023746	000046			.WORD	38	
3273	023750	007173			.WORD	EM21	
3274	023752	011546			.WORD	ERR4	
3275	023754				CKLOOP		
3276	023754	104006			EMT	C\$CLP1	
3277	023756	005037	002302		CLR	SAVCNT	;CLEAR BAD WORD COUNTER
3278	023762	005037	002232		CLR	CHECK	;CLEAR PRINT HEADER INDICATOR
3279	023766	012704	003500		MOV	#BUF1,R4	;GOOD DATA STORED IN BUF1
3280	023772	012702	004500		MOV	#BUF2,R2	;DATA BUFFER WRITTEN INTO BY MAINT.
3281	023776	012701	000377		MOV	#255,R1	
3282	024002	011437	002234		MOV	(R4),GDDAT	;EXPECTED DATA
3283	024006	011237	002236		MOV	(R2),BDDAT	;GET DATA FROM BUFFER
3284	024012	023737	002234	002236	CMP	GDDAT,BDDAT	
3285	024020	001440			BEQ	7\$	;DATA COMPARE
3286	024022	010237	002224		MOV	R2,TEMP0	;DATA ERR-GET ADDRESS
3287	024026	005237	002302		INC	SAVCNT	;INC BAD WORD COUNTER
3288	024032	005737	002232		TST	CHECK	;CHECK IF FIRST TIME
3289	024036	001007			BNE	6\$	
3290	024040				ERRDF	39.EM22,ERR3	
3291	024040	104462			TRAP	T\$ERRCODE	
3292	024042	000047			.WORD	39	
3293	024044	007255			.WORD	EM22	
3294	024046	011444			.WORD	ERR3	

CVRLAA.P11 14-APR-78 15:04

TEST 36 - TEST RLV11 MAINT. FUNCTION - FLAG MODE

3295	024050	005237	002232		INC	CHECK	;PRINT HEADER ONCE
3296	024054	000422			BR	7\$	
3297	024056			6\$:	PRINTX	#FRMT14,E,BA,E.DA,TMPO,GDDAT,BDDAT	
3298	024056	013746	002236		MOV	BDDAT,-(SP)	
3299	024062	013746	002234		MOV	GDDAT,-(SP)	
3300	024066	013746	002224		MOV	TMPO,-(SP)	
3301	024072	013746	002164		MOV	E.DA,-(SP)	
3302	024076	013746	002162		MOV	E.BA,-(SP)	
3303	024102	012746	013217		MOV	#FRMT14,-(SP)	
3304	024106	012746	000006		MOV	#6,-(SP)	
3305	024112	010600			MOV	SP,RO	
3306	024114	104015			EMT	C\$PNTX	
3307	024116	062706	000016		ADD	#16,SP	
3308	024122			7\$:	CKLOOP		
3309	024122	104006			EMT	C\$CLP1	
3310	024124	005722			TST	(R2)+	;INCREMENT BUFFER
3311	024126	005724			TST	(R4)+	;INCREMENT BUFFER
3312	024130	005301			DEC	R1	;FINISHED BUFFER?
3313	024132	001323			BNE	5\$	;RETURN FOR NEXT COMPARE
3314	024134	005737	002232		TST	CHECK	;CHECK FOR ERROR HEADER FLAG
3315	024140	001412			BEQ	77\$	
3316	024142				PRINTB	#FRMT98,SAVCNT	;PRINT BAD WORD COUNT
3317	024142	013746	002302		MOV	SAVCNT,-(SP)	
3318	024146	012746	012572		MOV	#FRMT98,-(SP)	
3319	024152	012746	000002		MOV	#2,-(SP)	
3320	024156	010600			MOV	SP,RO	
3321	024160	104014			EMT	C\$PNTB	
3322	024162	062706	000006		ADD	#6,SP	
3323	024166	012737	123456	002234	77\$:	MOV	#123456,GDDAT
3324	024174	011237	002236		MOV	(R2),BDDAT	;EXPECTED DATA IN LAST WORD+1
3325	024200	023737	002234	002236	MOV	GDDAT,BDDAT	;GET LAST WORD+1 FROM BUF2
3326	024206	001404			CMP	GDDAT,BDDAT	
3327	024210				BEQ	8\$	
3328	024210	104462			ERRDF	40,EM23,ERR4	
3329	024212	000050			TRAP	T\$EACODE	
3330	024214	007344			.WORD	40	
3331	024216	011546			.WORD	EM23	
3332	024220			8\$:	.WORD	ERR4	
3333	024220	104006			CKLOOP		
3334	024222				EMT	C\$CLP1	
3335	024222			10000\$:	ENDSEG		
3336	024222	104005			EMT	C\$ESEG	
3337	024224	005723			TST	(R3)+	;INC CRC PATTERN
3338	024226	062737	000002	002300	ADD	#2,PATSAV	;UPDATE PATTERN TABLE
3339	024234	020327	002604		CMP	R3,#CRCEND	;CHECK FOR END
3340	024240	001402			BEQ	10\$	;END OF TEST
3341	024242	000137	023462		JMP	101\$	;CONTINUE TEST
3342							
3343	024246			10\$:			
3344				ENDTST			
3345	024246			L10064:			
3346	024246			EMT	C\$ETST		
3347	024246	104001					
3348							
3349							
3350							

.SBTTL **TEST 37** - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

CVRLAA.P11 14-APR-78 15:04

TEST 37 - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

BGNTST

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

STARS

```

;*****
;PERFORM RLV11 MAINTENANCE FUNCTION 0 (INT. MODE) AND CHECK
;FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
;WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER
;THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
;RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
;FIFO INTO BUF2 MEMORY FOR PROPER DATA.
;CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
;VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.

```

STARS

;*****

```

TST T.CNTRL ;RLV11?
BEQ 100$ ;YES,RLV11
JMP 10$ ;NO,SKIP TEST
100$: MOV #PATCRC,R3 ;GET CRC PATTERN
MOV #PATDAT,PATSAV ;GET DATA PATTERN
101$: MOV (R3),102$
MOV #PATSAV,103$
JSR R5,CALCRC ;CALCULATE CRC BEFORE TEST
102$: .WORD 0 ;PATTERN FOR CRC TEST
BGNSEG
EMT CSBSEG
JSR R5,SETPAT ;SETUP PATTERN IN BUFFER
103$: .WORD 0 ;BUFFER PATTERN
SETPRI #PRI00 ;SET PRIORITY TO ZERO
MOV #PRI00,R0
EMT CSSPRI
CLR INTFLG ;CLEAR INT. FLAG
JSR R5,LDFUN ;PERFORM MAINT. FUNCTION
MAINT!INTEN ;MAINT FUNCTION INT. DRIVEN
-511. ;WORD COUNT
MATINT ;MESSAGE
JSR R5,WTCRDY ;WAIT FOR READY
CKLOOP
EMT CSCLP1
SETPRI #PRI07
MOV #PRI07,R0
EMT CSSPRI
TST INTFLG
BNE 104$
ERRDF 41,EM24,ERR0
TRAP TSEACODE
104$: .WORD 41
.WORD EM24
.WORD ERR0
CLR INTFLG ;CLEAR INT. FLAG
CKLOOP
EMT CSCLP1
JSR R5,CHERR ;CHECK CONTROLLER FOR ERRORS
CKLOOP
EMT CSCLP1
MOV #BUF1+1776,GDDAT

```

```

3351
3352 024250
3353
3354 024250
3355
3356
3357
3358
3359
3360
3361
3362
3363
3364 024250
3365
3366 024250 005737 002260
3367 024254 001402
3368 024256 000137 025124
3369 024262 012703 002514
3370 024266 012737 002606 002300
3371 024274 011337 024312
3372 024300 017737 155774 024322
3373 024306 004537 015216
3374 024312 000000
3375 024314
3376 024314 104004
3377 024316 004537 015454
3378 024322 000000
3379 024324
3380 024324 012700 000000
3381 024330 104041
3382 024332 005037 002176
3383 024336 004537 015542
3384 024342 000100
3385 024344 177001
3386 024346 006104
3387 024350 004537 016354
3388 024354
3389 024354 104006
3390 024356
3391 024356 012700 000340
3392 024362 104041
3393 024364 005737 002176
3394 024370 001004
3395 024372
3396 024372 104462
3397 024374 000051
3398 024376 007404
3399 024400 011352
3400 024402 005037 002176
3401 024406
3402 024406 104006
3403 024410 004537 014302
3404 024414
3405 024414 104006
3406 024416 012737 005476 002234

```

CVRLAA.P11

14-APR-78 15:04

TEST 37 - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

3407	024424	013737	002162	002236	MOV	E.BA,BDDAT	
3408	024432	023737	002234	002236	CMP	GDDAT,BDDAT	;TEST BA REGISTER
3409	024440	001404			BEQ	1\$	
3410	024442				ERRDF	42,EM10,ERR4	;DATA WRONG IN BA REGISTER
3411	024442	104462			TRAP	T\$ERRCODE	
3412	024444	000052			.WORD	42	
3413	024446	006566			.WORD	EM10	
3414	024450	011546			.WORD	ERR4	
3415	024452				1\$: CKLOOP		;CHECK FOR LOOP MODE
3416	024452	104006			EMT	C\$CLP1	
3417	024454	013737	002152	002234	MOV	B.DA,GDDAT	;GET BEFORE DA REGISTER
3418	024462	013737	002164	002236	MOV	E.DA,BDDAT	
3419	024470	005037	002222		CLR	TEMP1	
3420	024474	113737	002152	002222	MOVB	B.DA,TEMP1	
3421	024502	062737	000006	002222	ADD	#6,TEMP1	;+6 TO DA LOW BYTE
3422	024510	113737	002222	002234	MOVB	TEMP1,GDDAT	;STORE LOW BYTE OF DA
3423	024516	023737	002234	002236	CMP	GDDAT,BDDAT	
3424	024524	001404			BEQ	2\$	
3425	024526				ERRDF	43,EM12,ERR4	
3426	024526	104462			TRAP	T\$ERRCODE	
3427	024530	000053			.WORD	43	
3428	024532	006670			.WORD	EM12	
3429	024534	011546			.WORD	ERR4	
3430	024536				2\$: CKLOOP		
3431	024536	104006			EMT	C\$CLP1	
3432	024540	013737	002242	002234	MOV	GDCRCA,GDDAT	;GET CRC OF DA+3 VALUE
3433	024546	013737	002166	002236	MOV	E.MP,BDDAT	;GET CONTROLLER CRC OF DA+3
3434	024554	023737	002234	002236	CMP	GDDAT,BDDAT	
3435	024562	001404			BEQ	3\$	
3436	024564				ERRDF	44,EM20,ERR4	
3437	024564	104462			TRAP	T\$ERRCODE	
3438	024566	000054			.WORD	44	
3439	024570	007120			.WORD	EM20	
3440	024572	011546			.WORD	ERR4	
3441	024574				3\$: CKLOOP		
3442	024574	104006			EMT	C\$CLP1	
3443	024576	013737	002244	002234	MOV	GDCRCB,GDDAT	
3444	024604	013737	002170	002236	MOV	E.MP1,BDDAT	
3445	024612	023737	002234	002236	CMP	GDDAT,BDDAT	
3446	024620	001404			BEQ	4\$	
3447	024622				ERRDF	45,EM21,ERR4	
3448	024622	104462			TRAP	T\$ERRCODE	
3449	024624	000055			.WORD	45	
3450	024626	007173			.WORD	EM21	
3451	024630	011546			.WORD	ERR4	
3452	024632				4\$: CKLOOP		
3453	024632	104006			EMT	C\$CLP1	
3454	024634	005037	002302		CLR	SAVCNT	;CLEAR BAD WORD COUNTER
3455	024640	005037	002232		CLR	CHECK	;CLEAR PRINT HEADER INDICATOR
3456	024644	012704	003500		MOV	#BUF1,R4	;GOOD DATA BUFFER
3457	024650	012702	004500		MOV	#BUF2,R2	;DATA BUFFER WRITTEN INTO BY MAINT.
3458	024654	012701	000377		MOV	#255,R1	
3459	024660	011537	002234		5\$: MOV	(R4),GDDAT	;EXPECTED DATA
3460	024664	011267	002236		MOV	(R2),BDDAT	;GET DATA FROM BUFFER
3461	024670	023737	002234	002236	CMP	GDDAT,BDDAT	
3462	024676	001440			BEQ	7\$	;DATA COMPARE

CVRLAA.P11 14-APR-78 15:04

TEST 37 - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

3463	024700	010237	002224		MOV	R2, TMPO	; DATA ERR-GET ADDRESS
3464	024704	005237	002302		INC	SAVCNT	; INC. BAD WORD COUNT
3465	024710	005737	002232		TST	CHECK	; CHECK IF FIRST TIME
3466	024714	001007			BNE	6\$	
3467	024716				ERRDF	46, EM22, ERR3	
3468	024716	104462			TRAP	TSEACODE	
3469	024720	000056			.WORD	46	
3470	024722	007255			.WORD	EM22	
3471	024724	011444			.WORD	ERR3	
3472	024726	005237	002232		INC	CHECK	; PRINT HEADER ONCE
3473	024732	000422			BR	7\$	
3474	024734			6\$:	PRINTX	#FRMT14, E.BA, E.DA, TMPO, GDDAT, BDDAT	
3475	024734	013746	002236		MOV	BDDAT, -(SP)	
3476	024740	013746	002234		MOV	GDDAT, -(SP)	
3477	024744	013746	002224		MOV	TMPO, -(SP)	
3478	024750	013746	002164		MOV	E.DA, -(SP)	
3479	024754	013746	002162		MOV	E.BA, -(SP)	
3480	024760	012746	013217		MOV	#FRMT14, -(SP)	
3481	024764	012746	000006		MOV	#6, -(SP)	
3482	024770	010600			MOV	SP, R0	
3483	024772	104015			EMT	CSPNTX	
3484	024774	062706	000016		ADD	#16, SP	
3485	025000			7\$:	CKLOOP		
3486	025000	104006			EMT	CSCLP1	
3487	025002	005722			TST	(R2)+	; INCREMENT BUFFER
3488	025004	005724			TST	(R4)+	; INCREMENT BUFFER
3489	025006	005301			DEC	R1	; FINISHED BUFFER?
3490	025010	001323			BNE	5\$	; RETURN FOR NEXT COMPARE
3491	025012	005737	002232		TST	CHECK	; CHECK ERROR HEADER FLAG
3492	025016	001412			BEQ	77\$	
3493	025020				PRINTB	#FRMT98, SAVCNT	; PRINT BAD WORD COUNT
3494	025020	013746	002302		MOV	SAVCNT, -(SP)	
3495	025024	012746	012572		MOV	#FRMT98, -(SP)	
3496	025030	012746	000002		MOV	#2, -(SP)	
3497	025034	010600			MOV	SP, R0	
3498	025036	104014			EMT	CSPNTB	
3499	025040	062706	000006		ADD	#6, SP	
3500	025044	012737	123456	002234	MOV	#123456, GDDAT	; EXPECTED DATA IN LAST WORD+1
3501	025052	011237	002236		MOV	(R2), BDDAT	; GET LAST WORD+1 FROM BUF2
3502	025056	023737	002234	002236	CMP	GDDAT, BDDAT	
3503	025064	001404			BEQ	8\$	
3504	025066				ERRDF	47, EM23, ERR4	
3505	025066	104462			TRAP	TSEACODE	
3506	025070	000057			.WORD	47	
3507	025072	007344			.WORD	EM23	
3508	025074	011546			.WORD	ERR4	
3509	025076			8\$:	CKLOOP		
3510	025076	104006			EMT	CSCLP1	
3511	025100				ENDSEG		
3512	025100			10000\$:			
3513	025100	104005			EMT	CSESEG	
3514	025102	005723			TST	(R3)+	; INC. CRC PATTERN
3515	025104	062737	000002	002300	ADD	#2, PATSAV	; UPDATE PATTERN TABLE
3516	025112	020327	002604		CMP	R3, #CRCEND	; CHECK FOR END
3517	025116	001402			BEQ	10\$	; END OF TEST
3518	025120	000137	024274		JMP	101\$	; CONTINUE TEST

CVRLAA.P11 14-APR-78 15:04

TEST 37 - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

```

3519
3520 025124
3521
3522 025124
3523 025124
3524 025124 104001
3525
3526
3527
3528 025126
3529
3530 025126
3531
3532
3533
3534
3535
3536 025126
3537
3538 025126 005737 002260
3539 025132 001402
3540 025134 000137 025472
3541 025140 005001
3542 025142 012702 000400
3543 025146 012703 003500
3544 025152 010123
3545 025154 005201
3546 025156 005302
3547 025160 001374
3548 025162 012702 000400
3549 025166 012703 004500
3550 025172 005023
3551 025174 005302
3552 025176 001375
3553 025200 005037 002176
3554 025204
3555 025204 012700 000000
3556 025210 104041
3557 025212 004537 015542
3558 025216 000100
3559 025220 177001
3560 025222 006104
3561 025224 004537 016354
3562 025230
3563 025230 104006
3564 025232
3565 025232 012700 000340
3566 025236 104041
3567 025240 005737 002176
3568 025244 001004
3569 025246
3570 025246 104462
3571 025250 000060
3572 025252 007404
3573 025254 011352
3574 025256 005037 002176

```

10\$:

ENDTST
L10065:

EMT CSETST

.SBTTL **TEST 38** - RLV11 FIFO ADDRESS TEST

BGNTST

STARS

```

*****
:TEST THAT FIFO OPERATES CORRECTLY.STORE ADDRESS PATTERN
:IN BUF1 (0-255) THAT CONTAINS A UNIQUE PATTERN IN EACH LOCATION.
:PERFORM MAINTENANCE FUNCTION AND TEST BUF2 FOR PROPER FIFO
:ADDRESSING.

```

STARS

```

*****

```

```

:;*****
:      TST      T.CNTR      ;RL11 OR RLV11
:      BEQ      1$          ;RLV11:PERFORM TEST
:      JMP      10$         ;RL11;SKIP TEST
1$:      CLR      R1
:      MOV      #255.,R2
:      MOV      #BUF1,R3    ;SETUP TO STORE PATTERN IN BUF1
2$:      MOV      R1,(R3)+
:      INC      R1          ;INC. PATTERN
:      DEC      R2
:      BNE      2$
:      MOV      #256.,R2    ;SETUP TO CLEAR BUF2
:      MOV      #BUF2,R3
3$:      CLR      (R3)+
:      DEC      R2
:      BNE      3$
:      CLR      INTFLG      ;CLEAR INT. FLAG
:      SETPRI   #PRI00
:      MOV      #PRI00,R0
:      EMT      C$SPRI
:      JSR      R5,LDGUN    ;LOAD FUNCTION
:      MAINT:INTEN          ;MAINT. WITH INTERRUPT
:      -511.
:      MATINT
:      JSR      R5,WTCRDY   ;WORD COUNT
:      CKLOOP
:      EMT      C$CLP1
:      SETPRI   #PRI07
:      MOV      #PRI07,R0  ;MAINT. MESSAGE
:      EMT      C$SPRI
:      TST      INTFLG     ;WAIT FOR READY
:      BNE      4$
:      ERDF     4$,EM24,ERRO
:      TRAP     T$EACODE
:      .WORD    4$
:      .WORD    EM24
:      .WORD    ERRO
4$:      CLR      INTFLG

```


CVRLAA.P11 14-APR-78 15:04

TEST 38 - RLV11 FIFO ADDRESS TEST

3575	025262		CKLOOP		
3576	025262	104006	EMT	C\$CLP1	
3577	025264	005037 002302	CLR	SAVCNT	; CLEAR BAD WORD COUNTER
3578	025270	005037 002232	CLR	CHECK	; CLEAR ERROR HEADER FLAG
3579	025274	005001	CLR	R1	
3580	025276	012702 000377	MOV	#255, R2	
3581	025302	012703 004500	MOV	#BUF2, R3	
3582	025306	010137 002234	5S: MOV	R1, GDDAT	; EXPECTED DATA
3583	025312	011337 002236	MOV	(R3), BDDAT	; DATA IN BUFFER
3584	025316	023737 002234 002236	CMP	GDDAT, BDDAT	
3585	025324	001440	BEQ	7S	
3586	025326	010337 002224	MOV	R3, TMPO	; GET ADDRESS FOR PRINTOUT
3587	025332	005237 002302	INC	SAVCNT	; INC. BAD WORD COUNTER
3588	025336	005737 002232	TST	CHECK	; CHECK ERROR HEADER FLAG
3589	025342	001007	BNE	6S	
3590	025344		ERRDF	49, EM25, ERR3	
3591	025344	104462	TRAP	T\$ERRCODE	
3592	025346	000061	.WORD	49	
3593	025350	007446	.WORD	EM25	
3594	025352	011444	.WORD	ERR3	
3595	025354	005237 002232	INC	CHECK	
3596	025360	000422	BR	7S	
3597	025362		6S: PRINTX	#FRMT14, E.BA, E.DA, TMPO, GDDAT, BDDAT	
3598	025362	013746 002236	MOV	BDDAT, -(SP)	
3599	025366	013746 002234	MOV	GDDAT, -(SP)	
3600	025372	013746 002224	MOV	TMPO, -(SP)	
3601	025376	013746 002164	MOV	E.DA, -(SP)	
3602	025402	013746 002162	MOV	E.BA, -(SP)	
3603	025406	012746 013217	MOV	#FRMT14, -(SP)	
3604	025412	012746 000006	MOV	#6, -(SP)	
3605	025416	010600	MOV	SP, R0	
3606	025420	104015	EMT	C\$PNTX	
3607	025422	062706 000016	ADD	#16, SP	
3608	025426		7S: CKLOOP		
3609	025426	104006	EMT	C\$CLP1	
3610	025430	005723	TST	(R3)+	
3611	025432	005201	INC	R1	; UPDATE PATTERN EXPECTED
3612	025434	005302	DEC	R2	
3613	025436	001323	BNE	5S	
3614	025440	005737 002232	TST	CHECK	; CHECK ERROR FLAG
3615	025444	001412	BEQ	10S	
3616	025446		PRINTB	#FRMT98, SAVCNT	; PRINT NUMBER OF BAD WORDS
3617	025446	013746 002302	MOV	SAVCNT, -(SP)	
3618	025452	012746 012572	MOV	#FRMT98, -(SP)	
3619	025456	012746 000002	MOV	#2, -(SP)	
3620	025462	010600	MOV	SP, R0	
3621	025464	104014	EMT	C\$PNTB	
3622	025466	062706 000006	ADD	#6, SP	
3623					
3624	025472		10S:		
3625					
3626	025472		ENDTST		
3627	025472		L10066:		
3628	025472	104001	EMT	C\$SETST	
3629					
3630			.SBTTL	**TEST 39** - RLV11 FIFO ADDRESS COMPLEMENT TEST	

CVRLAA.P11 14-APR-78 15:04

TEST 39 - RLV11 FIFO ADDRESS COMPLEMENT TEST

```

3631
3632 025474
3633
3634 025474
3635
3636
3637
3638
3639
3640 025474
3641
3642 025474 005737 002260
3643 025500 001402
3644 025502 000137 026044
3645 025506 012701 177777
3646 025512 012702 000400
3647 025516 012703 003500
3648 025522 010123
3649 025524 005301
3650 025526 005302
3651 025530 001374
3652 025532 012702 000400
3653 025536 012703 004500
3654 025542 005023
3655 025544 005302
3656 025546 001375
3657 025550 005037 002176
3658 025554
3659 025554 012700 000000
3660 025560 104041
3661 025562 004537 015542
3662 025566 000100
3663 025570 177001
3664 025572 006104
3665 025574 004537 016354
3666 025600
3667 025600 104006
3668 025602
3669 025602 012700 000340
3670 025606 104041
3671 025610 005737 002176
3672 025614 001004
3673 025616
3674 025616 104462
3675 025620 000062
3676 025622 007404
3677 025624 011352
3678 025626 005037 002176
3679 025632
3680 025632 104006
3681 025634 005037 002302
3682 025640 005037 002232
3683 025644 012701 177777
3684 025650 012702 000377
3685 025654 012703 004500
3686 025660 010137 002234

```

BGNTST

STARS

```

*****
:TEST THAT FIFO OPERATES CORRECTLY. STORE ADDRESS COMPLEMENT PAT.
:IN BUF1 (0-255) THAT CONTAINS A UNIQUE PATTERN IN EACH LOCATION.
:PERFORM MAINTENANCE FUNCTION AND TEST BUF2 FOR PROPER FIFO
:ADDRESSING.

```

STARS

```

*****

```

```

:*****
: TST T.CNTRL ;RL11 OR RLV11
: BEQ 1$ ;RLV11:PERFORM TEST
: JMP 10$ ;RL11:SKIP TEST
1$: MOV #177777,R1
: MOV #256.,R2
: MOV #BUF1,R3 ;SETUP TO STORE PATTERN IN BUF1
2$: MOV R1,(R3)+ ;NEXT COMP. PATTERN
: DEC R1
: DEC R2
: BNE 2$ ;SETUP TO CLEAR BUF2
3$: MOV #256.,R2
: MOV #BUF2,R3
: CLR (R3)+
: DEC R2
: BNE 3$
: CLR INTFLG ;CLEAR INT. FLAG
: SETPRI #PRI00
: MOV #PRI00,R0
: EMT CSSPRI
: JSR R5,LDFUN ;LOAD FUNCTION
: MAINT!INTEN ;MAINT. WITH INTERRUPT
: -511 ;WORD COUNT
: MATINT ;MAINT. MESSAGE
: JSR R5,WTCRDY ;WAIT FOR READY
: CKLOOP
: EMT CSCLP1
: SETPRI #PRI07
: MOV #PRI07,R0
: EMT CSSPRI
: TST INTFLG ;CHECK FOR INTERRUPT
: BNE 4$
: ERROF 50,EM24,ERRO
: TRAP TSEACODE
: .WORD 50
: .WORD EM24
: .WORD ERRO
4$: CLR INTFLG
: CKLOOP
: EMT CSCLP1
: CLR SAVCNT ;CLEAR BAD WORD COUNTER
: CLR CHECK ;CLEAR ERROR HEADER FLAG
: MOV #177777,R1
: MOV #255.,R2
: MOV #BUF2,R3
5$: MOV R1,GDOAT ;EXPECTED DATA

```


CVRLAA.P11 14-APR-78 15:04

TEST 39 - RLV11 FIFO ADDRESS COMPLEMENT TEST

```

3687 025664 011337 002236
3688 025670 023737 002234 002236
3689 025676 001440
3690 025700 010337 002224
3691 025704 005237 002302
3692 025710 005737 002232
3693 025714 001007
3694 025716
3695 025716 104462
3696 025720 000063
3697 025722 007505
3698 025724 011444
3699 025726 005237 002232
3700 025732 000422
3701 025734
3702 025734 013746 002236
3703 025740 013746 002234
3704 025744 013746 002224
3705 025750 013746 002164
3706 025754 013746 002162
3707 025760 012746 013217
3708 025764 012746 000006
3709 025770 010600
3710 025772 104015
3711 025774 062706 000016
3712 026000
3713 026000 104006
3714 026002 005723
3715 026004 005301
3716 026006 005302
3717 026010 001323
3718 026012 005737 002232
3719 026016 001412
3720 026020
3721 026020 013746 002302
3722 026024 012746 012572
3723 026030 012746 000002
3724 026034 010600
3725 026036 104014
3726 026040 062706 000006
3727
3728 026044
3729 026044
3730 026044
3731 026044 104001
3732
3733
3734
3735
3736 026046
3737
3738 026046
3739
3740
3741
3742

```

```

MOV (R3),BDDAT ;DATA IN BUFFER
CMP GDDAT,BDDAT
BEQ 7$
MOV R3, TMPO ;GET ADDRESS FOR PRINTOUT
INC SAVCNT ;INC. BAD WORD COUNTER
TST CHECK ;CHECK ERROR HEADER FLAG
BNE 6$
ERRDF 51, EM26, ERR3
TRAP TSEARCH
WORD 51
WORD EM26
WORD ERR3
INC CHECK
BR 7$
6$: PRINTX #FRMT14, E.BA, E.DA, TMPO, GDDAT, BDDAT
MOV BDDAT, -(SP)
MOV GDDAT, -(SP)
MOV TMPO, -(SP)
MOV E.DA, -(SP)
MOV E.BA, -(SP)
MOV #FRMT14, -(SP)
MOV #6, -(SP)
MOV SP, R0
EMT C$PNTX
ADD #16, SP
7$: CKLOOP
EMT C$CLP1
TST (R3)+
DEC R1 ;GET NEXT PATTERN
DEC R2
BNE 5$
TST CHECK ;CHECK ERROR FLAG
BEQ 10$
PRINTB #FRMT98, SAVCNT ;PRINT NO. OF BAD WORDS
MOV SAVCNT, -(SP)
MOV #FRMT98, -(SP)
MOV #2, -(SP)
MOV SP, R0
EMT C$PNTB
ADD #6, SP
10$:
ENDTST
L10067: EMT C$ETST

```

.SBTTL **TEST 40** - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE

BGNTST

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

STARS

```

;*****
;PERFORM RLV11 MAINT. FUNCTION WITH COMPLEMENT DATA PATTERNS IN BUF1
;CHECK FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
;WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER

```

CVRLAA.P11 14-APR-78 15:04

TEST 40 - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE

```

3743
3744
3745
3746
3747
3748 026046
3749
3750 026046 005737 002260
3751 026052 001402
3752 026054 000137 026722
3753 026060 012703 002514
3754 026064 012737 002606 002300
3755 026072 011337 026110
3756 026076 017737 154176 026120
3757 026104 004537 015216
3758 026110 000000
3759 026112
3760 026112 104004
3761 026114 004537 015642
3762 026120 000000
3763 026122
3764 026122 012700 000000
3765 026126 104041
3766 026130 005037 002176
3767 026134 004537 015542
3768 026140 000100
3769 026142 177001
3770 026144 006104
3771 026146 004537 016354
3772 026152
3773 026152 104006
3774 026154
3775 026154 012700 000340
3776 026160 104041
3777 026162 005737 002176
3778 026166 001004
3779 026170
3780 026170 104462
3781 026172 000064
3782 026174 007404
3783 026176 011352
3784 026200 005037 002176
3785 026204
3786 026204 104006
3787 026206 004537 014302
3788 026212
3789 026212 104006
3790 026214 012737 005476 002234
3791 026222 013737 002162 002236
3792 026230 023737 002234 002236
3793 026236 001404
3794 026240
3795 026240 104462
3796 026242 000065
3797 026244 006566
3798 026246 011546

```

```

:THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
:RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
:FIFO INTO BUF2 MEMORY FOR PROPER DATA.
:CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
:VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.
:STARS
:;*****

```

```

:;*****
TST T.CNTRL ;RLV11?
BEQ 100S ;YES,RLV11
JMP 10S ;NO,SKIP TEST
100S: MOV #PATCRC,R3 ;GET CRC PATTERN
MOV #PATDAT,PATSAV ;GET DATA PATTERN
101S: MOV (R3),102S
MOV #PATSAV,103S
JSR RS,CALCRC ;CALCULATE CRC BEFORE TEST
102S: .WORD 0 ;PATTERN FOR CRC TEST
BGNSEG
EMT CSBSEG
JSR RS,SETCMP ;SETUP PATTERN IN BUFFER
103S: .WORD 0 ;BUFFER PATTERN
SETPRI #PRI00 ;SET PRIORITY TO ZERO
MOV #PRI00,R0
EMT CSSPRI
CLR INTFLG ;CLEAR INT. FLAG
JSR RS,LDFUN ;PERFORM MAINT. FUNCTION
MAINT!INTEN ;MAINT FUNCTION INT. DRIVEN
-511. ;WORD COUNT
MATINT ;MESSAGE
JSR RS,WTCRDY ;WAIT FOR READY
CKLOOP
EMT CSCLP1
SETPRI #PRI07
MOV #PRI07,R0
EMT CSSPRI
TST INTFLG
BNE 104S
ERRDF 52,EM24,ERR0
TRAP TSEACODE
. WORD 52
. WORD EM24
. WORD ERR0
104S: CLR INTFLG ;CLEAR INT. FLAG
CKLOOP
EMT CSCLP1
JSR RS,CHERR ;CHECK CONTROLLER FOR ERRORS
CKLOOP
EMT CSCLP1
MOV #BUF1+1776,GDDAT
MOV E.BA,BODAT
CMP GDDAT,BDDAT ;TEST BA REGISTER
BEQ 1S
ERRDF 53,EM10,ERR4 ;DATA WRONG IN BA REGISTER
TRAP TSEACODE
. WORD 53
. WORD EM10
. WORD ERR4

```


CVRLAA.P11 14-APR-78 15:04

TEST 40 - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE

3799	026250				1\$: CKLOOP		;CHECK FOR LOOP MODE
3800	026250	104006			EMT	C\$CLP1	
3801	026252	013737	002152	002234	MOV	B.DA, GDDAT	;GET BEFORE DA REGISTER
3802	026260	013737	002164	002236	MOV	E.DA, BDDAT	
3803	026266	005037	002222		CLR	TEMP1	
3804	026272	113737	002152	002222	MOVB	B.DA, TEMP1	
3805	026300	062737	000006	002222	ADD	#6, TEMP1	;+6 TO DA LOW BYTE
3806	026306	113737	002222	002234	MOVB	TEMP1, GDDAT	;STORE LOW BYTE OF DA
3807	026314	023737	002234	002236	CMP	GDDAT, BDDAT	
3808	026322	001404			BEQ	2\$	
3809	026324				ERRDF	54. EM12, ERR4	
3810	026324	104462			TRAP	T\$EACODE	
3811	026326	000066			.WORD	54	
3812	026330	006670			.WORD	EM12	
3813	026332	011546			.WORD	ERR4	
3814	026334				2\$: CKLOOP		
3815	026334	104006			EMT	C\$CLP1	
3816	026336	013737	002242	002234	MOV	GDCRCA, GDDAT	;GET CRC OF DA+3 VALUE
3817	026344	013737	002166	002236	MOV	E.MP, BDDAT	;GET CONTROLLER CRC OF DA+3
3818	026352	023737	002234	002236	CMP	GDDAT, BDDAT	
3819	026360	001404			BEQ	3\$	
3820	026362				ERRDF	55. EM20, ERR4	
3821	026362	104462			TRAP	T\$EACODE	
3822	026364	000067			.WORD	55	
3823	026366	007120			.WORD	EM20	
3824	026370	011546			.WORD	ERR4	
3825	026372				3\$: CKLOOP		
3826	026372	104006			EMT	C\$CLP1	
3827	026374	013737	002244	002234	MOV	GDCRCB, GDDAT	
3828	026402	013737	002170	002236	MOV	E.MP1, BDDAT	
3829	026410	023737	002234	002236	CMP	GDDAT, BDDAT	
3830	026416	001404			BEQ	4\$	
3831	026420				ERRDF	56. EM21, ERR4	
3832	026420	104462			TRAP	T\$EACODE	
3833	026422	000070			.WORD	56	
3834	026424	007173			.WORD	EM21	
3835	026426	011546			.WORD	ERR4	
3836	026430				4\$: CKLOOP		
3837	026430	104006			EMT	C\$CLP1	
3838	026432	005037	002302		CLR	SAVCNT	;CLEAR BAD WORD COUNTER
3839	026436	005037	002232		CLR	CHECK	;CLEAR PRINT HEADER INDICATOR
3840	026442	012704	003500		MOV	#BUF1, R4	;GOOD DATA BUFFER
3841	026446	012702	004500		MOV	#BUF2, R2	;DATA BUFFER WRITTEN INTO BY MAINT.
3842	026452	012701	000377		MOV	#255, R1	
3843	026456	011437	002234		MOV	(R4), GDDAT	;EXPECTED DATA
3844	026462	011237	002236		MOV	(R2), BDDAT	;GET DATA FROM BUFFER
3845	026466	023737	002234	002236	CMP	GDDAT, BDDAT	
3846	026474	001440			BEQ	7\$	;DATA COMPARE
3847	026476	010237	002224		MOV	R2, TMP0	;DATA ERR-GET ADDRESS
3848	026502	005237	002302		INC	SAVCNT	;INC. BAD WORD COUNTER
3849	026506	005737	002232		TST	CHECK	;CHECK IF FIRST TIME
3850	026512	001007			BNE	6\$	
3851	026514				ERRDF	57. EM22, ERR3	
3852	026514	104462			TRAP	T\$EACODE	
3853	026516	000071			.WORD	57	
3854	026520	007255			.WORD	EM22	

CVRLAA.P11 14-APR-78 15:04

TEST 40 - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE

```

3855 026522 011444
3856 026524 005237 002232
3857 026530 000422
3858 026532
3859 026532 013746 002236
3860 026536 013746 002234
3861 026542 013746 002224
3862 026546 013746 002164
3863 026552 013746 002162
3864 026556 012746 013217
3865 026562 012746 000006
3866 026566 010600
3867 026570 104015
3868 026572 062706 000016
3869 026576
3870 026576 104006
3871 026600 005722
3872 026602 005724
3873 026604 005301
3874 026606 001323
3875 026610 005737 002232
3876 026614 001412
3877 026616
3878 026616 013746 002302
3879 026622 012746 012572
3880 026626 012746 000002
3881 026632 010600
3882 026634 104014
3883 026636 062706 000006
3884 026642 012737 123456 002234
3885 026650 011237 002236
3886 026654 023737 002234 002236
3887 026662 001404
3888 026664
3889 026664 104462
3890 026666 000072
3891 026670 007344
3892 026672 011546
3893 026674
3894 026674 104006
3895 026676
3896 026676
3897 026676 104005
3898 026700 005723
3899 026702 062737 000002 002300
3900 026710 020327 002604
3901 026714 001402
3902 026716 000137 026072
3903
3904 026722
3905 026722
3906 026722
3907 026722 104001
3908
3909
3910

```

```

WORD ERR3
INC CHECK ;PRINT HEADER ONCE
BR 7$
6$: PRINTX #FRMT14,E.BA,E.DA,TMPO,GDDAT,BDDAT
MOV BDDAT,-(SP)
MOV GDDAT,-(SP)
MOV TMPO,-(SP)
MOV E.DA,-(SP)
MOV E.BA,-(SP)
MOV #FRMT14,-(SP)
MOV #6,-(SP)
MOV SP,R0
EMT CSPNTX
ADD #16,SP
7$: CKLOOP
EMT CSCLP1
TST (R2)+ ;INCREMENT BUFFER
TST (R4)+ ;INCREMENT BUFFER
DEC R1 ;FINISHED BUFFER?
BNE 5$ ;RETURN FOR NEXT COMPARE
TST CHECK ;CHECK ERROR FLAG
BEQ 77$
PRINTB #FRMT98,SAVCNT ;PRINT NO OF BAD WORDS
MOV SAVCNT,-(SP)
MOV #FRMT98,-(SP)
MOV #2,-(SP)
MOV SP,R0
EMT CSPNTB
ADD #6,SP
77$: MOV #123456,GDDAT ;EXPECTED DATA IN LAST WORD+1
MOV (R2),BDDAT ;GET LAST WORD+1 FROM BUF2
CMP GDDAT,BDDAT
BEQ 8$
ERRDF 58,EM23,ERR4
TRAP TSERCODE
58
WORD 58
WORD EM23
WORD ERR4
8$: CKLOOP
EMT CSCLP1
ENDSEG
10000$: EMT CSESEG
TST (R3)+ ;INC. CRC PATTERN
ADD #2,PATSAV ;UPDATE PATTERN TABLE
CMP R3,#CRCEND ;CHECK FOR END
BEQ 10$ ;END OF TEST
JMP 101$ ;CONTINUE TEST
10$:
ENDTST
L10070: EMT CSETST
.SBTTL **TEST 41** - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

```


CVRLAA.P11 14-APR-78 15:04

TEST 41 - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

3911 026724

BGNTST

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

3912
3913 026724

STARS

```

*****
:PERFORM RLV11 MAINT. FUNCTION WITH RANDOM DATA PATTERNS IN BUF1
:RANDOM PATTERN IS THE SAME FOR EACH CONTROLLER UNDER TEST.
:RANDOM PATTERN WILL CHANGE AT END OF PASS.
:RANDOM PATTERN WILL INIT AT START OR RESTART.
:CHECK FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
:WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER
:THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
:RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
:FIFO INTO BUF2 MEMORY FOR PROPER DATA.
:CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
:VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.

```

3914
3915
3916
3917
3918
3919
3920
3921
3922
3923
3924
3925
3926 026724

STARS

;*****

3927 026724 005737 002260

```

:;*****
: TST T.CNTRL :RLV11?
: BEQ 101$ :YES,RLV11
: JMP 10$ :NO SKIP TEST
: MOV TEMLO,102$ :STARTING RANDOM PATTERN
: MOV TEMLO,LONUM :RESET RANDOM START
: JSR RS,CALCRC :CALCULATE CRC BEFORE TEST
: .WORD 0 :PATTERN FOR CRC TEST
: JSR RS,SETRAN :SETUP RANDOM PATTERN IN BUFFER

```

3928 026724 001402 002260

3929 026730 000137 027544

3930 026732 000137 002274

3931 026736 013737 002274

3932 026744 013737 015216

3933 026752 004537 015216

3934 026756 000000 015734

3935 026760 004537 015734

3936 026764 104004

3937 026764 104004

3938 026766 012700 000000

3939 026766 012700 000000

3940 026772 104041 002176

3941 026774 005037 015542

3942 027000 004537 015542

3943 027004 000100 016354

3944 027006 177001 016354

3945 027010 006104 016354

3946 027012 004537 016354

3947 027016 104006

3948 027016 104006

3949 027020 012700 000340

3950 027020 012700 000340

3951 027024 104041 002176

3952 027026 005737 002176

3953 027032 001004 002176

3954 027034 104462 002176

3955 027034 104462 002176

3956 027036 000073 014302

3957 027040 007404 014302

3958 027042 011352 002176

3959 027044 005037 002176

3960 027050 104006 002234

3961 027050 104006 002236

3962 027052 004537 002236

3963 027056 104006 002236

3964 027056 104006 002236

3965 027060 012737 002236

3966 027066 013737 002236

101\$:

102\$:

104\$:

```

:*****
: TST T.CNTRL :RLV11?
: BEQ 101$ :YES,RLV11
: JMP 10$ :NO SKIP TEST
: MOV TEMLO,102$ :STARTING RANDOM PATTERN
: MOV TEMLO,LONUM :RESET RANDOM START
: JSR RS,CALCRC :CALCULATE CRC BEFORE TEST
: .WORD 0 :PATTERN FOR CRC TEST
: JSR RS,SETRAN :SETUP RANDOM PATTERN IN BUFFER
: BGNSEG
: EMT CSBSEG
: SETPRI #PRI00 :SET PRIORITY TO ZERO
: MOV #PRI00,RO
: EMT CSSPRI
: CLR INTFLG :CLEAR INT. FLAG
: JSR RS,LDFUN :PERFORM MAINT. FUNCTION
: MAINT!INTEN :MAINT FUNCTION INT. DRIVEN
: -511. :WORD COUNT
: MATINT :MESSAGE
: JSR RS,WTCRDY :WAIT FOR READY
: CKLOOP
: EMT CSCLP1
: SETPRI #PRI07
: MOV #PRI07,RO
: EMT CSSPRI
: TST INTFLG
: BNE 104$
: ERROF 59,EM24,ERRO
: TRAP T$ERCODE
: .WORD 59
: .WORD EM24
: .WORD ERRO
: CLR INTFLG :CLEAR INT. FLAG
: CKLOOP
: EMT CSCLP1
: JSR RS,CHERR :CHECK CONTROLLER FOR ERRORS
: CKLOOP
: EMT CSCLP1
: MOV #BUF1+1776,GDDAT
: MOV E.BA,BDDAT

```

F08

PAGE: 0096AS
SEQ 0096

CVRLAA.P11 14-APR-78 15:04

TEST 41 - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

3967	027074	023737	002234	002236	CMP	GDDAT,BDDAT	;TEST BA REGISTER
3968	027102	001404			BEQ	1\$	
3969	027104				ERRDF	60, EM10,ERR4	;DATA WRONG IN BA REGISTER
3970	027104	104462			TRAP	T\$ERRCODE	
3971	027106	000074			.WORD	60	
3972	027110	006566			.WORD	EM10	
3973	027112	011546			.WORD	ERR4	
3974	027114				1\$: CKLOOP		;CHECK FOR LOOP MODE
3975	027114	104006			EMT	CSCLP1	
3976	027116	013737	002152	002234	MOV	B.DA,GDDAT	;GET BEFORE DA REGISTER
3977	027124	013737	002164	002236	MOV	E.DA,BDDAT	
3978	027132	005037	002222		CLR	TEMP1	
3979	027136	113737	002152	002222	MOVB	B.DA,TEMP1	
3980	027144	062737	000006	002222	ADD	#6,TEMP1	;+6 TO DA LOW BYTE
3981	027152	113737	002222	002234	MOVB	TEMP1,GDDAT	;STORE LOW BYTE OF DA
3982	027160	023737	002234	002236	CMP	GDDAT,BDDAT	
3983	027166	001404			BEQ	2\$	
3984	027170				ERRDF	61, EM12,ERR4	
3985	027170	104462			TRAP	T\$ERRCODE	
3986	027172	000075			.WORD	61	
3987	027174	006670			.WORD	EM12	
3988	027176	011546			.WORD	ERR4	
3989	027200				2\$: CKLOOP		
3990	027200	104006			EMT	CSCLP1	
3991	027202	013737	002242	002234	MOV	GDCRCA,GDDAT	;GET CRC OF DA+3 VALUE
3992	027210	013737	002166	002236	MOV	E.MP,BDDAT	;GET CONTROLLER CRC OF DA+3
3993	027216	023737	002234	002236	CMP	GDDAT,BDDAT	
3994	027224	001404			BEQ	3\$	
3995	027226				ERRDF	62, EM20,ERR4	
3996	027226	104462			TRAP	T\$ERRCODE	
3997	027230	000076			.WORD	62	
3998	027232	007120			.WORD	EM20	
3999	027234	011546			.WORD	ERR4	
4000	027236				3\$: CKLOOP		
4001	027236	104006			EMT	CSCLP1	
4002	027240	013737	002244	002234	MOV	GDCRCB,GDDAT	
4003	027246	013737	002170	002236	MOV	E.MP1,BDDAT	
4004	027254	023737	002234	002236	CMP	GDDAT,BDDAT	
4005	027262	001404			BEQ	4\$	
4006	027264				ERRDF	63, EM21,ERR4	
4007	027264	104462			TRAP	T\$ERRCODE	
4008	027266	000077			.WORD	63	
4009	027270	007173			.WORD	EM21	
4010	027272	011546			.WORD	ERR4	
4011	027274				4\$: CKLOOP		
4012	027274	104006			EMT	CSCLP1	
4013	027276	005037	002302		CLR	SAVCNT	;CLEAR BAD WORD COUNTER
4014	027302	005037	002232		CLR	CHECK	;CLEAR PRINT HEADER INDICATOR
4015	027306	012703	003500		MOV	#BUF1,R3	;BUFFER WITH RANDOM NUMBERS
4016	027312	012702	004500		MOV	#BUF2,R2	;DATA BUFFER WRITTEN INTO BY MAINT.
4017	027316	012701	000377		MOV	#255,R1	
4018	027322	011337	002234		MOV	(R3),GDDAT	;EXPECTED DATA
4019	027326	011237	002236		MOV	(R2),BDDAT	;GET DATA FROM BUFFER
4020	027332	023737	002234	002236	CMP	GDDAT,BDDAT	
4021	027340	001440			BEQ	7\$	;DATA COMPARE
4022	027342	010237	002224		MOV	R2,TEMP0	;DATA ERR-GET ADDRESS

CVRLAA.P11 14-APR-78 15:04

TEST 41 - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

4023	027346	005237	002302	INC	SAVCNT	;INC BAD WORD COUNT
4024	027352	005737	002232	TST	CHECK	;CHECK IF FIRST TIME
4025	027356	001007		BNE	65	
4026	027360			ERRDF	64. EM22,ERR3	
4027	027360	104462		TRAP	TSERCODE	
4028	027362	000100		.WORD	64	
4029	027364	007255		.WORD	EM22	
4030	027366	011444		.WORD	ERR3	
4031	027370	005237	002232	INC	CHECK	;PRINT HEADER ONCE
4032	027374	000422		BR	75	
4033	027376			65:	PRINTX	#FRMT14,E.BA,E.DA,TMPO,GDDAT,BDDAT
4034	027376	013746	002236	MOV	BDDAT,-(SP)	
4035	027402	013746	002234	MOV	GDDAT,-(SP)	
4036	027406	013746	002224	MOV	TMPO,-(SP)	
4037	027412	013746	002164	MOV	E.DA,-(SP)	
4038	027416	013746	002162	MOV	E.BA,-(SP)	
4039	027422	012746	013217	MOV	#FRMT14,-(SP)	
4040	027426	012746	000006	MOV	#6,-(SP)	
4041	027432	010600		MOV	SP,R0	
4042	027434	104015		EMT	CSPNTX	
4043	027436	062706	000016	ADD	#16,SP	
4044	027442			75:	CKLOOP	
4045	027442	104006		EMT	C\$CLP1	
4046	027444	005722		TST	(R2)+	;INCREMENT BUFFER
4047	027446	005723		TST	(R3)+	;INCREMENT GOOD BUFFER
4048	027450	005301		DEC	R1	;FINISHED BUFFER?
4049	027452	001323		BNE	55	;RETURN FOR NEXT COMPARE
4050	027454	005737	002232	TST	CHECK	;CHECK ERROR FLAG
4051	027460	001412		BEQ	775	
4052	027462			PRINTB	#FRMT98,SAVCNT	;PRINT NO. OF BAD WORDS
4053	027462	013746	002302	MOV	SAVCNT,-(SP)	
4054	027466	012746	012572	MOV	#FRMT98,-(SP)	
4055	027472	012746	000002	MOV	#2,-(SP)	
4056	027476	010600		MOV	SP,R0	
4057	027500	104014		EMT	CSPNTB	
4058	027502	062706	000006	ADD	#6,SP	
4059	027506	012737	123456	775:	MOV	#123456,GDDAT
4060	027514	011237	002236	MOV	(R2),BDDAT	;EXPECTED DATA IN LAST WORD+1
4061	027520	023737	002234	002236	CMP	GDDAT,BDDAT
4062	027526	001404		BEQ	85	
4063	027530			ERRDF	65. EM23,ERR4	
4064	027530	104462		TRAP	TSERCODE	
4065	027532	000101		.WORD	65	
4066	027534	007344		.WORD	EM23	
4067	027536	011546		.WORD	ERR4	
4068	027540			85:	CKLOOP	
4069	027540	104006		EMT	C\$CLP1	
4070	027542			ENDSEG		
4071	027542			10000\$:	EMT	C\$ESEG
4072	027542	104005				
4073				10\$:	NOP	;NOP IS NEEDED TO INSURE THAT "LASTAD"
4074	027544	000240				;WILL HAVE BIT 7 CLEARED FOR APT
4075						;COMPATIBILITY.
4076						
4077	027546			ENDTST		
4078	027546			L10071:		

H08

PAGE: 0098AS
SEQ 0098

CVRLAA.P11 14-APR-78 15:04

TEST 41 - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

4079	027546	104001				EMT	CSETST
4080							
4081	027550					BGNMOD	HRDPRM
4082							
4083	027550					BGNHRD	
4084	027550	000025				.WORD	L10072-LSHARD/2
4085							
4086	027552					GPRML	CNTMSG,CNT,1,YES
4087	027552	004130				.WORD	TSCODE
4088	027554	027640				.WORD	CNTMSG
4089	027556	000001				.WORD	1
4090	027560					GPRMA	CSRMSG,CSR,0,160000,177776,YES
4091	027560	000031				.WORD	TSCODE
4092	027562	027624				.WORD	CSRMSG
4093	027564	160000				.WORD	TSLOLIM
4094	027566	177776				.WORD	TSHILIM
4095	027570					GPRMA	VECMMSG,VECT,0,0,776,YES
4096	027570	001031				.WORD	TSCODE
4097	027572	027656				.WORD	VECMMSG
4098	027574	000000				.WORD	TSLOLIM
4099	027576	000776				.WORD	TSHILIM
4100	027600					GPRMD	BRMSG,PRIOR,0,340,0,7,YES
4101	027600	002032				.WORD	TSCODE
4102	027602	027645				.WORD	BRMSG
4103	027604	000340				.WORD	340
4104	027606	000000				.WORD	TSLOLIM
4105	027610	000007				.WORD	TSHILIM
4106	027612					GPRMD	DRMSG,DRBT,0,03400,0,7,YES
4107	027612	003032				.WORD	TSCODE
4108	027614	027665				.WORD	DRMSG
4109	027616	003400				.WORD	03400
4110	027620	000000				.WORD	TSLOLIM
4111	027622	000007				.WORD	TSHILIM
4112							
4113	027624					ENDHRD	
4114						.EVEN	
4115	027624					L10072:	
4116							
4117	027624	052502	020123	042101	CSRMSG:	.ASCIZ	/BUS ADDRESS/
4118	027632	051104	051505	000123			
4119	027640	046122	030461	000	CNTMSG:	.ASCIZ	/RL11/
4120	027645	102	020122	042514	BRMSG:	.ASCIZ	/BR LEVEL/
4121	027652	042526	000114				
4122	027656	042526	052103	051117	VECMMSG:	.ASCIZ	/VECTOR/
4123	027664	000					
4124	027665	104	044522	042526	DRMSG:	.ASCIZ	/DRIVE/
4125	027672	000					
4126		027674				.EVEN	
4127							
4128	027674					ENDMOD	
4129							
4130							
4131							
4132	027674				BGNMOD	SFTPRM	
4133							
4134	027674				BGNSFT		

CVRLAA.P11 14-APR-78 15:04

TEST 41 - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

```

4135 027674 000014 .WORD L10073-LSSOFT/2
4136 027676 .GPRML DMSG,DLT,1,YES
4137 027676 000130 .WORD TSCODE
4138 027700 027726 .WORD DMSG
4139 027702 000001 .WORD 1
4140 027704 XFERF 15
4141 027704 006044 .WORD TSCODE
4142 027706 .GPRMD EMSG,ELT,0,177777,0,177777,YES
4143 027706 001032 .WORD TSCODE
4144 027710 027763 .WORD EMSG
4145 027712 177777 .WORD 177777
4146 027714 000000 .WORD TSLOLIM
4147 027716 177777 .WORD TSHILIM
4148 027720 15: .GPRML SMSG,SIZE,1,YES
4149 027720 002130 .WORD TSCODE
4150 027722 027752 .WORD SMSG
4151 027724 000001 .WORD 1
4152 027726 ENDSFT
4153 .EVEN
4154 027726 L10073:
4155
4156 027726 051104 050117 047440 DMSG: .ASCIZ /DROP ON ERROR LIMIT/
      027752 052501 047524 044523 SMSG: .ASCIZ /AUTOSIZE/
      027763 105 051122 051117 EMSG: .ASCIZ /ERROR LIMIT/

4157 030000 .EVEN
4158
4159 030000 ENDMOD
4160
4161 030114 .=30114
4162 ;AREA RESERVED AS PATCH AREA FOR DIAGNOSTIC
4163 ;.=30114 WAS SELECTED TO PROVIDE AN ACT MAILBOX ADDRESS
4164 ;IN THE SUPERVISOR WITH THE PARITY BIT CLEARED.
4165 030114 LASTAD
4166 .EVEN
4167 030114 L$LAST::
4168
4169 ;FOR LSI-11 APT COMPATIBILITY MAKE SURE THAT THE "LASTAD" ADDRESS
4170 ;HAS A CLEAR BIT 7. WHEN APT TRANSMITS THE MAILBOX ADDRESS TO
4171 ;THE LSI-11, THE LSI ODT WILL CLEAR BIT 7 CREATING AN INVALID
4172 ;MAILBOX ADDRESS IF THAT BIT WAS SET.
4173

```

J08

CVRLAA.SUP

11-APR-78 09:27

DIAGNOSTIC SUPERVISOR -- LOW CORE SET UP

PAGE: 0100AS
SEQ 0100

4174
4175 060710 000000
060712 000000
060714 000000
060716 000000
060722
000200

.SBTTL DIAGNOSTIC SUPERVISOR -- LOW CORE SET UP
.WORD 0 ;SPACE FOR USER POOL POINTER
.WORD 0 ;SIZE
.WORD 0 ;CHECKSUM (NOT CURRENTLY USED)
.WORD 0 ;SIZE OF H.W. PTAB. ALLOCATION
END.SUPV=+.2
.END 200

K08

PAGE: 0101AS
SEQ 0101

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

L08

PAGE: 0102AS
SEQ 0102

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MO8

PAGE: 0103AS
SEQ 0103

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

PAGE: 0104AS
SEQ 0104

CROSS REFERENCE TABLE -- USER SYMBOLS

	3996	4007	4027	4064										
CSEHR= 000003	8#													
CSERSF= 000001	8#	1498	1541	1583	1625	1666	1709	1751	1793					
CSERSO= 000004	8#													
CSESCA= 000010	8#	1956	2005	2051	2102	2153	2201	2246	2290	2335	2762	2889	2894	
CSESEG= 000005	2910													
	8#	1966	2015	2061	2112	2161	2209	2254	2298	2343	2919	2970	3018	
CSESUB= 000003	3075													
CSETST= 000001	8#	3182	3336	3513	3897	4072								
	8#													
	8#	1506	1549	1591	1633	1674	1717	1759	1801	1847	1882	1914	1969	
	2018													
	2628													
	2699													
	2732													
	3731													
	8#	3907	4079											
CSEXIT= 000032	8#													
CSGMAN= 000043	8#													
CSGPHR= 000042	8#	818												
CSGPRI= 000040	8#													
CSGTIN= 000052	8#	3145												
CSINIT= 000011	8#	884												
CSINLP= 000020	8#	946												
CSKWOF= 000035	8#													
CSKWON= 000034	8#													
CSLOOP= 000100	8#													
CSMANI= 000051	8#													
CSMSG = 000023	8#	453	462	478	504	524	533	551	568	583	600			
CSPNTB= 000014	8#	473	488	499	518	546	561	578	595	610	623	633	644	
	655	664	671	680	860	3321	3498	3621	3725	3882	4057			
CSPNTF= 000017	8#	959												
CSPNTS= 000016	8#													
CSPNTX= 000015	8#	3306	3483	3606	3710	3867	4042							
CSPOIN= 000040	8#	1049	1428	1450	1498	1541	1583	1625	1666	1709	1751	1793	1840	
	1874	1906	1951	2000	2046	2097	2148	2196	2241	2285	2330	2386	2420	
	2452	2492	2506	2548	2563	2605	2619	2662	2676	2689	2770	2784	2798	

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

EM1	006315	439#	1500	1668						
EM10	006566	439#	3236	3413	3797	3972				
EM101	011114	439#	660							
EM102	011161	439#	667	676	1007	1119				
EM11	006627	439#								
EM12	006670	439#	3251	3428	3812	3987				
EM13	006731	439#	2844							
EM14	006763	439#	2772							
EM15	007011	439#	2786							
EM16	007037	439#	2800							
EM17	007065	439#	2906							
EM2	006342	439#	1543	1711						
EM20	007120	439#	3262	3439	3823	3998				
EM21	007173	439#	3273	3450	3834	4009				
EM22	007255	439#	3293	3470	3854	4029				
EM23	007344	439#	3330	3507	3891	4066				
EM24	007404	439#	3068	3139	3398	3572	3676	3782	3957	
EM25	007446	439#	3593							
EM26	007505	439#	3697							
EM27	007557	439#	2964							
EM3	006367	439#	1585	1753						
EM30	007633	439#	3012							
EM31	007707	439#	3166							
EM4	006414	439#	1627	1795						
EM44	007730	439#	2664							
EM45	007763	439#	2678							
EM46	010016	439#	2691							
EM5	006441	439#	1953							
EM6	006512	439#	2002							
EM61	010051	439#	2099							
EM62	010132	439#	2150							
EM63	010215	439#	2198							
EM64	010276	439#	2243							
EM65	010361	439#	2287							
EM66	010442	439#	2332							
EM67	010525	439#	1842	2388						
EM7	006540	439#	2048							
EM70	010562	439#	1876	2422						
EM71	010617	439#	1908	2454						
EM72	010654	439#	2494							
EM73	010707	439#	2508							
EM74	010742	439#	2550							
EM75	010774	439#	2565							
EM76	011026	439#	2607							
EM77	011061	439#	2621							
END	014100	838	853	868#						
ENDDAT	002676	392#								
ENDPAT	002512	329#	2010	2056	2205	2250	2294	2339		
END.OF	036470	4175#								
END.SU=	060722	4175#*								
ENVIRO	030166	4175#*								
EOP.CH	057370	4175#								
EOP.FM	032734	4175#								
EOP.IN	035100	4175#*								
ERR =	100000	148#	1828	1944	2090	2141	2374	2535	2592	2648
ERRFOR	043714	4175#								

G
G

CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

FRMT5 012660
FRMT6 012731
FRMT98 012572
FRMT99 012655
FSAU = 000015
FSBGN = 000040

FSCLEA= 000007
FSDU = 000016
FSEND = 000041

FSHARD= 000004
FSHW = 000013
FSINIT= 000006
FSJMP = 000050
FSMOD = 000000

FMSG = 000011

FSPWR = 000017
FSRPT = 000012
FSSEG = 000003

FSOFT= 000005
FSSRV = 000010
FSSUB = 000002
FSSW = 000014
FSTEST= 000001

684#	558	684#	3495	3618	3722	3879	4054							
684#	543	575	684#											
8#	931	935												
8#	17	63		76	192	196	435	437	440	444	446	456	465	
482	509	527	536	554	571	586	688	690	703	705	717	719		
765	768	770	887	889	891	913	917	919	927	929	931	939		
944	1401	1464	1470	1505	1512	1548	1555	1590	1597	1632	1638	1673		
1680	1716	1723	1758	1765	1800	1846	1853	1881	1888	1913	1920	1968		
1936	1956	1968	1975	1988	2005	2017	2024	2036	2051	2063	2070	2082		
2102	2114	2121	2133	2153	2163	2170	2183	2201	2211	2218	2230	2246		
2256	2263	2275	2290	2300	2307	2319	2335	2345	2352	2392	2399	2427		
2434	2459	2466	2514	2521	2569	2576	2627	2633	2698	2704	2731	2738		
2762	2806	2813	2854	2861	2877	2889	2894	2910	2925	2932	2951	2974		
2980	2999	3022	3028	3048	3079	3086	3107	3186	3193	3216	3346	3353		
3376	3523	3529	3627	3633	3730	3737	3760	3906	3912	3937	4078	4082		
4084	4129	4133	4135	4160										
8#	891	909												
8#	919	923												
8#	17	63		76	192	196	435	437	440	444	454	463	479	
505	525	534	552	569	584	601	688	690	703	705	717	719		
765	768	885	887	889	911	913	917	925	927	929	937	939		
944	1407	1464	1470	1505	1507	1512	1548	1550	1555	1590	1592	1597		
1632	1634	1638	1673	1675	1680	1716	1718	1723	1758	1760	1765	1800		
1802	1806	1846	1848	1853	1881	1883	1888	1913	1915	1920	1956	1967		
1968	1970	1975	2005	2016	2017	2019	2024	2051	2062	2063	2065	2070		
2102	2113	2114	2116	2121	2153	2162	2163	2165	2170	2201	2210	2211		
2213	2218	2246	2255	2256	2258	2263	2290	2299	2300	2302	2307	2335		
2344	2345	2347	2352	2392	2394	2399	2427	2429	2434	2459	2461	2466		
2514	2516	2521	2569	2571	2576	2627	2629	2633	2698	2700	2704	2731		
2733	2738	2762	2806	2808	2813	2854	2856	2861	2889	2894	2910	2920		
2925	2927	2932	2971	2974	2976	2980	3019	3022	3024	3028	3076	3079		
3081	3086	3183	3186	3188	3193	3337	3346	3348	3353	3514	3523	3525		
3529	3627	3629	3633	3730	3732	3737	3898	3906	3908	3912	4073	4078		
4080	4082	4116	4129	4133	4155	4160								
8#	4084	4114	4141											
8#	692	700												
8#	770	883												
8#														
8#	17	63	76	192	196	435	437	440	444	688	690	703		
705	717	719	765	768	887	889	913	917	927	929	939	944		
1464	4082	4129	4133	4160										
8#	446	452	456	461	465	477	482	503	509	523	527	532		
536	550	554	567	571	582	586	599							
8#														
8#														
8#	1936	1965	1988	2014	2036	2060	2082	2111	2133	2160	2183	2208		
2230	2253	2275	2297	2319	2342	2877	2918	2951	2969	2999	3017	3048		
3074	3107	3181	3216	3335	3376	3512	3760	3896	3937	4071				
8#	4135	4141	4153											
8#	1401	1405												
8#														
8#	707	714												
8#	1470	1505	1512	1548	1555	1590	1597	1632	1638	1673	1680	1716		
1723	1758	1765	1800	1806	1846	1853	1881	1888	1913	1920	1968	1975		

CROSS REFERENCE TABLE -- USER SYMBOLS

		2017	2024	2063	2070	2114	2121	2163	2170	2211	2218	2256	2263	2300
		2307	2345	2352	2392	2399	2427	2434	2459	2466	2514	2521	2569	2576
		2627	2633	2698	2704	2731	2738	2806	2813	2854	2861	2925	2932	2974
		2980	3022	3028	3079	3086	3186	3193	3346	3353	3523	3529	3627	3633
		3730	3737	3906	3912	4078								
GARBAG	050744	4175#												
GCRCPT	002240	238#	1172*	1173	1254									
GDATEP	002250	242#	1275*	1276	1277*									
GDCRCA	002242	239#	1182*	3255	3432	3816	3991							
GDCRCB	002244	240#	1195*	3266	3443	3827	4002							
GDDAT	002234	236#	469	492	514	1495*	1538*	1581*	1622*	1663*	1706*	1749*	1790*	1825*
		1828#	1836	1867*	1899*	1939*	1940*	1941	1944*	1947	1990*	1993*	1994	1996
		2038#	2039	2042	2085*	2086*	2090*	2093	2136*	2137*	2141*	2144	2186*	2189*
		2192	2233*	2234*	2237	2278*	2281	2322*	2323*	2326	2371*	2374*	2382	2413*
		2445*	2488*	2502*	2532*	2535*	2536	2544	2559*	2589*	2592*	2593	2601	2615*
		2645*	2648*	2649	2658	2672*	2686*	2752*	2766	2778*	2780	2792*	2794	2874*
		2897	3229*	3231	3240*	3245*	3246	3255*	3257	3266*	3268	3282*	3284	3299
		3323*	3325	3406*	3408	3417*	3422*	3423	3432*	3434	3443*	3445	3459*	3461
		3476	3500*	3502	3582*	3584	3599	3686*	3688	3703	3790*	3792	3801*	3806*
		3807	3816*	3818	3827*	3829	3843*	3845	3860	3884*	3886	3965*	3967	3976*
		3981*	3982	3991*	3993	4002*	4004	4018*	4020	4035	4059*	4061		
		241#	1226*	1229	1272*	1275								
GDDATP	002246	4175#												
GETCHR	047450	4175#												
GETCMN	053034	4175#												
GETPAR	044526	4175#												
GETSWI	052030	4175#												
GET.TW	051600	4175#												
GLBDAT	002122	196#												
GLBEQA	002122	76#												
GLBERR	011352	444#												
GLBSUB	014216	944#												
GLBTXT	005500	437#												
GODRVR=	000202	168#												
GSBIT =	000002	170#												
GSTAT =	000004	163#	996	1069										
GSEXCP=	000400	8#												
GSHILI=	000002	8#												
GSLOLI=	000001	8#												
GSNO =	000000	8#												
GSOFFS=	000400	8#	4087	4091	4096	4101	4107	4137	4143	4149				
GSOFSI=	000376	8#	4087	4091	4096	4101	4107	4137	4143	4149				
GSPRMA=	000001	8#	4091	4096										
GSPRMD=	000002	8#	4101	4107	4143									
GSPRML=	000000	8#	4087	4137	4149									
GSRADA=	000140	8#												
GSRADB=	000000	8#												
GSRADD=	000040	8#												
GSRADF=	000200	8#												
GSRADL=	000120	8#	4087	4137	4149									
GSRADO=	000020	8#	4091	4096	4101	4107	4143							
GSRADT=	000100	8#												
GSXFER=	000004	8#	4141											
GSYES =	000010	8#	4087	4091	4096	4101	4107	4137	4143	4149				
HCORED	034656	4175#												
HCOREQ	034566	4175#												
HCORET	030402	4175#*												

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

HCRME	005560	439#	1028															
HC.ADR	030152	G 4175#																
HC.DEF	030144	G 4175#																
HC.DIA	030142	G 4175#																
HORBUF	003000	429#																
HORLST	015022	1080	1101#															
HERTZ.	034526	4175#																
HINUM	002270	250#	794#	800	828*	1325	1333	1338*										
HNFMS	005566	439#	1032	1131														
HOLDSP=	000020	4175#																
HPTCOO	013312	G 690#																
HROPRM	027550	G 4082#																
HW.ADR	030150	G 4175#																
HSARB	054206	4175#																
INITIT	030372	G 4175#																
INITCO	013462	G 768#																
INITIA	047420	4175#																
INIT.M	035504	4175#																
INIT.R	030206	G 4175#																
INPUTA	050346	4175#																
INTEN =	000100	147#	902	1078	1086	2837	2885	3054	3129	3384	3558	3662	3768	3943				
INTFLG	002176	221#	1402*	1409*	2832*	2839	2846*	2879*	2900	2913	3052*	3063	3070*	3121*				
		3134	3141*	3382*	3393	3400*	3553*	3567	3574*	3657*	3671	3678*	3766*	3777				
		3784*	3941*	3952	3959*													
INTFOR	043644	4175#																
INTSRV	016272	870	1402#	3175														
INVAL.	034452	4175#																
INVINT	043502	4175#																
INV.SM	031504	4175#																
IN.SUF	036454	4175#																
ISAU =	000041	8#	931#	937#														
ISCLN =	000041	8#	891#	911#														
ISDU =	000041	8#	919#	925#														
ISHRD =	000041	4084#	4116#															
ISINIT=	000041	8#	770#	885#														
ISMOD =	000041	8#	17#	63#	76#	192#	196#	435#	437#	440#	444#	688#	690#	703#				
		705#	717#	719#	765#	768#	887#	889#	913#	917#	927#	929#	939#	944#				
		1464#	4082#	4129#	4133#	4160#												
ISMSG =	000041	8#	446#	454#	456#	463#	465#	479#	482#	505#	509#	525#	527#	534#				
		536#	552#	554#	569#	571#	584#	586#	601#									
ISPR =	000041	8#																
ISRPT =	000041	8#																
ISSEG =	000041	8#	1470	1512	1555	1597	1638	1680	1723	1765	1806	1853	1888	1920				
		1936#	1956	1967#	1975	1988#	2005	2016#	2024	2036#	2051	2062#	2070	2082#				
		2102	2113#	2121	2133#	2153	2162#	2170	2183#	2201	2210#	2218	2230#	2246				
		2255#	2263	2275#	2290	2299#	2307	2319#	2335	2344#	2352	2399	2434	2466				
		2521	2576	2633	2704	2738	2813	2861	2877#	2910	2920#	2932	2951#	2971#				
		2980	2999#	3019#	3028	3048#	3076#	3086	3107#	3183#	3193	3216#	3337#	3353				
		3376#	3514#	3529	3633	3737	3760#	3898#	3912	3937#	4073#							
		4135#	4155#															
ISSFT =	000041	8#	1401#	1407#														
ISSRV =	000041	8#	1470	1512	1555	1597	1638	1680	1723	1765	1806	1853	1888	1920				
ISSUB =	000041	8#	2024	2070	2121	2170	2218	2263	2307	2352	2399	2434	2466	2521				
		1975	2024	2070	2121	2170	2218	2263	2307	2352	2399	2434	2466	2521				
		2576	2633	2704	2738	2813	2861	2932	2980	3028	3086	3193	3353	3529				
		3633	3737	3912														
ISTST =	000041	8#	1470#	1505#	1507#	1512#	1548#	1550#	1555#	1590#	1592#	1597#	1632#	1634#				

H09

PAGE: 0111AS
SEQ 0111

CVRLAA.SUP

11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

1638#	1673#	1675#	1680#	1716#	1718#	1723#	1758#	1760#	1765#	1800#	1802#	1806#
1846#	1848#	1853#	1881#	1883#	1888#	1913#	1915#	1920#	1968#	1970#	1975#	2017#
2019#	2024#	2063#	2065#	2070#	2114#	2116#	2121#	2163#	2165#	2170#	2211#	2213#
2218#	2256#	2258#	2263#	2300#	2302#	2307#	2345#	2347#	2352#	2392#	2394#	2399#
2427#	2429#	2434#	2459#	2461#	2466#	2514#	2516#	2521#	2569#	2571#	2576#	2627#
2629#	2633#	2698#	2700#	2704#	2731#	2733#	2738#	2762#	2806#	2808#	2813#	2854#
2856#	2861#	2889#	2894#	2925#	2927#	2932#	2974#	2976#	2980#	3022#	3024#	3028#
3079#	3081#	3086#	3186#	3188#	3193#	3346#	3348#	3353#	3523#	3525#	3529#	3627#
3629#	3633#	3730#	3732#	3737#	3906#	3908#	3912#	4078#	4080#			
8#												
4175#*												
4175#*												
222#	1064#	1076#	1077	1086	1091#	1092#	1093	1249#	1255#	1256#	1257#	1258
2478#	2481											
1249#	2952	3000	3053	3128	3219	3383	3557	3661	3767	3942		
1064#	2721	2754	2836	2884								
439#	1051											
4175#*												
447	457	466	482	510	528	537	555	571	586	603#	862	961
448	483	511	539	572	587	614#						
538	659#											
573	675#											
4175#*												
4175#												
251#	795#	799	827#	1301	1324	1331	1337#	3932#				
4175#*												
4175#*												
4175#												
4175#												
4175#												
4175#												
4175#*												
39#												
55	931#											
55#												
60#												
60	891#											
37#												
28#												
58#												
51#												
40	722#											
53	68#											
53#												
54#												
54	68#											
40#												
56	919#											
56#												
51	71#											
48#												
38#												
41#												
42#												
43#												
31	4084	4085#										

JSJMP = 000167
KBPTR 030224 G
KBUF 030226 G
LDCSR 002200
LDFUN 015542
LDFUNC 014602
LF 005676
LINE.F 030442 G
LINE1 012042
LINE2 012076
LINE3 012314
LINE4 012366
LOAD.F 035102
LOGMSG 047412
LONUM 002272
LPBFR 030222 G
LPCNTR 030220 G
LPT.AD 034544
LPT.RE 034540
LSI.RE 034534
LUP 055506
LUP.AD 042512
LSAPT 002036 G
LSAU 014212 G
LSAUT 002074 G
LSCCP 002106 G
LSCLEA 014144 G
LSCO 002032 G
LSDEPO 002011 G
LSDESC 002102 G
LSDEVP 002064 G
LSDISP 013340 G
LSDR 002112 G
LSORCT 002070 G
LSORS 002072 G
LSORST 002112 G
LSDTP 002040 G
LSDU 014206 G
LSDUT 002076 G
LSDVTY 002114 G
LSEF 002056 G
LSEFLG 002034 G
LSEXP1 002042 G
LSEXP2 002044 G
LSEXP3 002046 G
LSHARD 027552 G

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

LSHPCP	002016	G	31#		
LSHPTP	002022	G	33#		
LSHW	013314	G	33#	692	693#
LSICP	002104	G	59#		
LSINIT	013462	G	59#	770#	
LSLADP	002026	G	35#		
LSLAST	030114	G	35#	4167#	
LSMREV	002050	G	44#		
LSNAME	002000	G	19#		
LSREPP	002066	G	52#		
LSREV	002010	G	27#		
LSSOFT	027676	G	32#	4135	4136#
LSSPC	002062	G	50#		
LSSPCP	002020	G	32#		
LSSPTP	002024	G	34#		
LSSTA	002030	G	36#		
LSSW	013330	G	34#	707	708#
LSTIML	002014	G	30#		
LSTIMU	002054	G	47#		
LSTIMI	002052	G	46#		
LSTSTI	002100	G	57#		
LSUNIT	002012	G	29#	797	812
L.CLK.	034512		4175#		
L10000	011366		452#		
L10001	011400		461#		
L10002	011442		477#		
L10003	011544		503#		
L10004	011612		523#		
L10005	011624		532#		
L10006	011666		550#		
L10007	011724		567#		
L10010	011766		582#		
L10011	012040		599#		
L10012	013326		692	700#	
L10013	013336		707	714#	
L10014	014142		883#		
L10015	014204		909#		
L10016	014210		923#		
L10017	014214		935#		
L10020	016276		1405#		
L10021	016526		1505#		
L10022	016524		1548#		
L10023	016722		1590#		
L10024	017020		1632#		
L10025	017114		1673#		
L10026	017210		1716#		
L10027	017304		1758#		
L10030	017400		1800#		
L10031	017510		1846#		
L10032	017562		1881#		
L10033	017620		1913#		
L10034	017740		1968#		
L10035	020042		2017#		
L10036	020130		2063#		
L10037	020254		2114#		
L10040	020400		2163#		

CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

PAGE: 0114AS
SEQ 0114

CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

SUPFLA 030360 G
SUPV.T 030532 G
SUP.PR 031470
SVCCNT= 177777

SVCGBL= 000000

SVCHAN 037406
SVCINS= 000000

4175**	17	63	76	192	196	435	437	440	444	446	452	456
4175**	461	465	477	482	503	523	527	532	536	550	554	567
4175**	571	582	586	599	688	690	700	703	705	707	714	717
8#	719	765	768	770	883	887	891	909	913	917	919	923
	927	929	931	935	939	944	1401	1405	1464	1505	1512	1548
1555	1590	1597	1632	1638	1673	1680	1716	1723	1758	1765	1800	1806
1846	1853	1881	1888	1913	1920	1936	1965	1968	1975	1988	2014	2017
2024	2036	2060	2063	2070	2082	2111	2114	2121	2133	2160	2163	2170
2183	2208	2211	2218	2230	2253	2256	2263	2275	2297	2300	2307	2319
2342	2345	2352	2392	2399	2427	2434	2459	2466	2514	2521	2569	2576
2527	2533	2598	2704	2731	2738	2806	2813	2854	2861	2877	2918	2925
2932	2951	2969	2974	2980	2999	3017	3022	3028	3048	3074	3079	3086
3107	3181	3186	3193	3216	3335	3346	3353	3376	3512	3523	3529	3627
3633	3730	3737	3760	3896	3906	3912	3937	4071	4078	4082	4084	4114
4129	4133	4135	4153	4160								
8#	37	38	39	27	29	30	31	32	33	34	35	36
	52	53	54	40	42	43	44	46	47	48	50	51
437	444	446	456	465	482	509	527	536	554	571	586	596
693	705	708	719	722	768	770	889	891	917	919	929	931
944	4082	4085	4133	4136	4167#	4168						
4175**	19	20	21	22	23	24	25	26	27	28	29	30
8#	31	32	33	34	35	36	37	39	40	41	42	43
	44	45	46	47	48	49	50	52	53	54	55	56
462	463	468	469	470	471	472	473	474	475	478	479	485
486	487	488	489	490	491	492	493	494	495	496	497	498
499	500	501	504	505	513	514	515	516	517	518	519	520
524	525	533	534	535	544	545	546	547	548	551	552	557
558	559	560	561	562	563	568	569	575	576	577	578	579
580	583	584	589	590	593	598	599	594	595	596	597	600
601	604	605	606	607	608	609	610	611	612	615	616	617
618	619	620	621	622	623	624	625	626	627	628	629	630
631	632	633	634	635	636	637	638	639	640	641	642	643
644	645	646	647	648	649	650	651	652	653	654	655	656
657	660	661	662	663	664	665	666	667	668	669	670	671
672	673	676	677	678	679	680	681	682	692	693	707	708
721	722	723	724	725	726	727	728	729	730	731	732	733
734	735	736	737	738	739	740	741	742	743	744	745	746
747	748	749	750	751	752	753	754	755	756	757	758	759
760	761	762	763	772	773	774	775	776	777	778	779	780
781	782	783	784	785	786	787	788	789	790	791	792	793
804	805	806	807	808	817	818	819	820	821	842	843	844
845	846	847	848	850	851	852	856	857	858	859	860	861
862	864	865	866	867	868	869	870	871	872	873	874	875
877	878	879	880	884	885	893	894	895	900	901	902	905
906	907	910	911	924	925	936	937	946	947	948	949	956
957	958	959	960	961	963	964	965	966	967	1049	1050	1051
1052	1053	1406	1407	1411	1412	1422	1423	1424	1428	1429	1430	1431
1432	1442	1443	1444	1450	1451	1452	1453	1454	1482	1483	1484	1485
1486	1487	1488	1491	1492	1493	1498	1499	1500	1501	1502	1503	1504
1506	1507	1525	1526	1527	1528	1529	1530	1531	1534	1535	1536	1541

1542	1543	1544	1545	1546	1547	1549	1550	1567	1568	1569	1570	1571
1572	1573	1576	1577	1578	1583	1584	1585	1586	1587	1588	1589	1591
1592	1609	1610	1611	1612	1613	1614	1615	1618	1619	1620	1625	1626
1627	1628	1629	1630	1631	1633	1634	1650	1651	1652	1653	1654	1655
1656	1659	1660	1661	1666	1667	1668	1669	1670	1671	1672	1674	1675
1693	1694	1695	1696	1697	1698	1699	1702	1703	1704	1709	1710	1711
1712	1713	1714	1715	1717	1718	1735	1736	1737	1738	1739	1740	1741
1744	1745	1746	1751	1752	1753	1754	1755	1756	1757	1759	1760	1777
1778	1779	1780	1781	1782	1783	1786	1787	1788	1793	1794	1795	1796
1797	1798	1799	1801	1802	1822	1823	1824	1831	1832	1840	1841	1842
1843	1844	1847	1848	1869	1870	1874	1875	1876	1877	1878	1882	1883
1901	1902	1906	1907	1908	1909	1910	1914	1915	1936	1937	1951	1952
1953	1954	1955	1956	1957	1958	1966	1967	1969	1970	1988	1989	2000
2001	2002	2003	2004	2005	2006	2007	2015	2016	2018	2019	2036	2037
2046	2047	2048	2049	2050	2051	2052	2053	2061	2062	2064	2065	2082
2083	2097	2098	2099	2100	2101	2102	2103	2104	2112	2113	2115	2116
2133	2134	2148	2149	2150	2151	2152	2153	2154	2155	2161	2162	2164
2165	2183	2184	2196	2197	2198	2199	2200	2201	2202	2203	2209	2210
2225	2227	2228	2231	2241	2242	2243	2244	2245	2246	2247	2248	2254
2255	2257	2258	2275	2276	2278	2279	2287	2288	2289	2290	2291	2292
2298	2299	2301	2302	2319	2320	2321	2328	2333	2333	2334	2335	2336
2337	2343	2344	2345	2346	2347	2348	2370	2377	2378	2386	2387	2388
2389	2390	2393	2394	2415	2416	2421	2422	2423	2424	2428	2429	2429
2447	2448	2453	2454	2455	2456	2457	2460	2461	2492	2493	2494	2495
2496	2497	2498	2506	2507	2508	2509	2510	2515	2516	2548	2549	2550
2551	2552	2553	2554	2563	2564	2565	2566	2567	2570	2571	2605	2606
2607	2608	2609	2610	2611	2612	2613	2621	2622	2623	2628	2629	2662
2663	2664	2665	2666	2667	2668	2669	2677	2678	2679	2680	2681	2682
2689	2690	2691	2696	2697	2698	2700	2725	2726	2732	2733	2758	2759
2762	2763	2764	2770	2771	2772	2773	2774	2776	2777	2784	2785	2786
2787	2788	2790	2791	2798	2799	2800	2801	2802	2807	2808	2834	2835
2836	2842	2843	2844	2845	2846	2848	2849	2855	2856	2877	2878	2881
2882	2883	2889	2890	2891	2892	2895	2896	2897	2904	2906	2907	2908
2910	2911	2912	2919	2920	2921	2927	2935	2951	2958	2959	2962	2963
2964	2965	2966	2967	2968	2970	2971	2975	2976	2999	3000	3006	3007
3010	3011	3012	3013	3014	3015	3016	3018	3019	3023	3024	3048	3049
3050	3051	3052	3059	3060	3061	3062	3063	3066	3067	3068	3069	3070
3072	3073	3075	3076	3080	3081	3107	3108	3109	3110	3111	3112	3113
3114	3115	3116	3117	3118	3119	3120	3121	3133	3134	3137	3138	3139
3140	3141	3143	3144	3145	3146	3147	3157	3158	3159	3164	3165	3166
3167	3168	3169	3170	3171	3172	3173	3174	3175	3176	3177	3178	3179
3180	3182	3183	3187	3188	3189	3217	3225	3226	3228	3229	3234	3235
3236	3237	3238	3239	3240	3243	3250	3251	3252	3253	3254	3255	3260
3261	3262	3263	3264	3265	3266	3271	3272	3273	3274	3275	3276	3277
3291	3292	3293	3294	3295	3298	3299	3300	3301	3302	3303	3304	3305
3306	3307	3308	3309	3310	3311	3317	3319	3320	3321	3322	3323	3328
3329	3330	3331	3332	3333	3334	3335	3336	3337	3347	3348	3377	3380
3381	3382	3389	3390	3391	3392	3393	3396	3397	3398	3399	3400	3402
3403	3405	3406	3411	3412	3413	3414	3415	3416	3417	3426	3427	3428
3429	3430	3431	3432	3437	3438	3439	3440	3441	3442	3443	3448	3449
3450	3451	3452	3453	3454	3458	3459	3470	3471	3472	3475	3476	3477
3478	3479	3480	3481	3482	3483	3484	3485	3486	3487	3494	3495	3496
3497	3498	3499	3500	3505	3506	3507	3508	3509	3510	3511	3513	3514
3524	3525	3555	3556	3557	3563	3564	3565	3566	3567	3570	3571	3572
3573	3574	3576	3577	3591	3592	3593	3594	3595	3598	3599	3600	3601
3602	3603	3604	3605	3606	3607	3608	3609	3610	3617	3618	3619	3620

CVRLAA.SUP

11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

3621	3622	3623	3628	3629	3659	3660	3661	3667	3668	3669	3670	3671
3674	3675	3676	3677	3678	3680	3681	3695	3696	3697	3698	3699	3702
3703	3704	3705	3706	3707	3708	3709	3710	3711	3712	3713	3714	3721
3722	3723	3724	3725	3726	3727	3731	3732	3760	3761	3764	3765	3766
3773	3774	3775	3776	3777	3780	3781	3782	3783	3784	3786	3787	3789
3790	3795	3796	3797	3798	3799	3800	3801	3810	3811	3812	3813	3814
3815	3816	3821	3822	3823	3824	3825	3826	3827	3832	3833	3834	3835
3836	3837	3838	3852	3853	3854	3855	3856	3859	3860	3861	3862	3863
3864	3865	3866	3867	3868	3869	3870	3871	3878	3879	3880	3881	3882
3883	3884	3889	3890	3891	3892	3893	3894	3895	3897	3898	3907	3908
3937	3938	3939	3940	3941	3948	3949	3950	3951	3952	3955	3956	3957
3958	3959	3961	3962	3964	3965	3966	3971	3972	3973	3974	3975	3976
3985	3986	3987	3988	3989	3990	3991	3996	3997	3998	3999	4000	4001
4002	4007	4008	4009	4010	4011	4012	4013	4027	4028	4029	4030	4031
4034	4035	4036	4037	4038	4039	4040	4041	4042	4043	4044	4045	4046
4053	4054	4055	4056	4057	4058	4059	4064	4065	4066	4067	4068	4069
4070	4072	4073	4079	4080	4084	4085	4087	4088	4089	4090	4091	4092
4093	4094	4095	4096	4097	4098	4099	4100	4101	4102	4103	4104	4105
4106	4107	4108	4109	4110	4111	4112	4114	4115	4135	4136	4137	4138
4139	4140	4141	4142	4143	4144	4145	4146	4147	4148	4149	4150	4151
4152	4153	4154	4166	4167								
8#	17	63	76	192	196	435	437	440	444	446	452	456
461	465	477	482	503	509	523	527	532	536	550	554	567
571	582	586	599	688	690	692	700	703	705	707	714	717
719	765	768	770	883	887	889	891	909	913	917	919	923
927	929	931	935	939	944	1401	1405	1464	1470	1505	1512	1548
1555	1590	1597	1632	1638	1673	1680	1716	1723	1758	1765	1800	1806
1846	1853	1881	1888	1913	1920	1936	1957	1965	1968	1975	1988	2006
2014	2017	2024	2036	2052	2060	2063	2070	2082	2103	2111	2114	2121
2133	2154	2160	2163	2170	2183	2202	2208	2211	2218	2230	2247	2253
2256	2263	2275	2291	2297	2300	2307	2319	2336	2342	2345	2352	2392
2399	2427	2434	2459	2466	2514	2521	2559	2576	2627	2633	2698	2704
2731	2738	2806	2813	2854	2861	2877	2911	2918	2925	2932	2951	2969
2974	2980	2999	3017	3022	3028	3048	3074	3079	3086	3107	3181	3186
3193	3216	3335	3346	3353	3376	3512	3523	3529	3627	3633	3730	3737
3760	3896	3906	3912	3937	4071	4078	4082	4084	4114	4129	4133	4135
4141	4153	4160										
8#	8#											
533	550	452	453	461	462	477	478	503	504	523	524	532
883	884	551	567	568	582	583	599	600	700	701	714	715
1478	1505	909	910	923	924	935	936	1405	1406	1471	1472	1477
1590	1591	1506	1515	1516	1521	1522	1548	1549	1556	1557	1562	1563
1674	1683	1598	1599	1604	1605	1632	1633	1639	1640	1645	1646	1673
1766	1767	1684	1689	1690	1716	1717	1724	1725	1730	1731	1758	1759
1856	1860	1772	1773	1800	1801	1808	1809	1818	1819	1846	1847	1855
1930	1931	1861	1881	1882	1890	1891	1895	1896	1913	1914	1924	1925
2018	2026	1965	1966	1968	1969	1977	1978	1983	1984	2014	2015	2017
2111	2112	2027	2031	2032	2060	2061	2063	2064	2071	2072	2077	2078
2173	2178	2114	2115	2123	2124	2128	2129	2160	2161	2163	2164	2172
2256	2257	2179	2208	2209	2211	2212	2220	2221	2225	2226	2253	2254
2315	2342	2265	2266	2270	2271	2297	2335	2336	2330	2309	2310	2314
2406	2407	2343	2345	2346	2354	2355	2441	2459	2392	2393	2401	2402
2476	2514	2427	2428	2436	2437	2441	2442	2570	2460	2468	2469	2475
2627	2628	2515	2522	2523	2529	2530	2569	2570	2579	2580	2586	2587
2732	2740	2636	2637	2642	2643	2698	2699	2708	2709	2714	2715	2731
		2741	2744	2745	2806	2807	2815	2816	2826	2827	2854	2855

SVCSTK= 177777

SVCSUB= 177777
SVCTAG= 000000

11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

SVCTST= 177777

SWCHAN	034700	
SWITCH	053552	
SW.ADR	030154	G
SW.PTA	034664	
SYS.FT	043432	
SSL SYM=	010000	

TEMHI	002276
TEMLO	002274
TEMP1	002222

TEMP2	002212	
TEMP3	002214	
TEMP4	002216	
TEMPS	002220	
TERMI	055576	
TERMLI	053400	
TERMTA	047362	
TEST.M	035020	
TIMFLG	030350	G
TIMSRV	016300	
TIM.CO	030202	G
TIM.OP	044006	
TIMFNC	002262	
TIMPO	002224	

TMP1	002226
TMP2	002230
TOO.MA	047342
TRPFLG	002174

TRPHAN	016264
TST. AB	037330
TST. TO	031532
TYPEC	047736
TYPEPC	043632
TYPFLA	053254
TYPLIN	047634
TYPNUM	047216
TYPSTR	047654

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

TYP.ER 043462
TY.UNI 036474
TSARGC= 000002TSCODE= 002130
TSERCO= 000062

TSERRN= 000101

TSEXCP= 000000
TSFLAG= 000040
TSHILI= 177777
TSLOLI= 000000
TSLSYM= 010000TSMCAL= 177777
TSNEST= 177777

TSNSKO= 000000

TSNSK1= 000005

4175#													
4175#													
19#	20#	21#	22#	23#	24#	468#	474	485#	489	491#	500	513#	
519	543#	547	557#	562	575#	579	589#	596	604#	611	615#	624	
626#	634	636#	645	647#	656	660#	665	667#	672	676#	681	856#	
861	956#	960	3298#	3307	3317#	3322	3475#	3484	3494#	3499	3598#	3607	
3617#	3622	3702#	3711	3721#	3726	3859#	3868	3878#	3883	4034#	4043	4053#	
4058													
4087#	4091#	4096#	4101#	4107#	4137#	4141#	4143#	4149#					
1049#	1428#	1450#	1498#	1541#	1583#	1625#	1666#	1709#	1751#	1793#	1840#	1874#	
1906#	1951#	2000#	2046#	2097#	2148#	2196#	2241#	2285#	2330#	2386#	2420#	2452#	
2492#	2506#	2548#	2563#	2605#	2619#	2662#	2676#	2689#	2770#	2784#	2798#	2842#	
2904#	2962#	3010#	3066#	3137#	3164#	3234#	3249#	3260#	3271#	3291#	3328#	3396#	
3411#	3426#	3437#	3448#	3468#	3505#	3570#	3591#	3674#	3695#	3780#	3795#	3810#	
3821#	3832#	3852#	3889#	3955#	3970#	3985#	3996#	4007#	4027#	4064#			
8#	1050#	1429#	1451#	1499#	1542#	1584#	1626#	1667#	1710#	1752#	1794#	1841#	
1875#	1907#	1952#	2001#	2047#	2098#	2149#	2197#	2242#	2286#	2331#	2387#	2421#	
2453#	2493#	2507#	2549#	2564#	2606#	2620#	2663#	2677#	2690#	2771#	2785#	2799#	
2843#	2905#	2963#	3011#	3067#	3138#	3165#	3235#	3250#	3261#	3272#	3292#	3329#	
3397#	3412#	3427#	3438#	3449#	3469#	3506#	3571#	3592#	3675#	3696#	3781#	3796#	
3811#	3822#	3833#	3853#	3890#	3956#	3971#	3986#	3997#	4008#	4028#	4065#		
4091#	4095#	4096#	4100	4101#	4106	4107#	4112	4143#	4148				
1956#	2005#	2051#	2102#	2153#	2201#	2246#	2290#	2335#	2762#	2889#	2894#	2910#	
4091#	4094	4096#	4099	4101#	4105	4107#	4111	4143#	4147				
4091#	4093	4096#	4098	4101#	4104	4107#	4110	4143#	4146				
8#	453	462	478	504	524	533	551	568	583	600	701	715	
884	910	924	936	1406	1506	1549	1591	1633	1674	1717	1759	1801	
1847	1882	1914	1969	2018	2064	2115	2164	2212	2257	2301	2346	2393	
2428	2460	2515	2570	2628	2699	2732	2807	2855	2926	2975	3023	3080	
3187	3347	3524	3628	3731	3907	4079	4116	4155					
1#	8												
8#	17#	63#	76#	192#	196#	435#	437#	440#	444#	446#	452#	456#	
461#	465#	477#	482#	503#	509#	523#	527#	532#	536#	550#	554#	567#	
571#	582#	586#	599#	688#	690#	692#	700#	703#	705#	707#	714#	717#	
719#	765#	768#	770#	883#	887#	889#	891#	909#	913#	917#	919#	923#	
927#	929#	931#	935#	939#	944#	1401#	1405#	1464#	1470#	1505#	1512#	1548#	
1555#	1590#	1597#	1632#	1638#	1673#	1680#	1716#	1723#	1758#	1765#	1800#	1806#	
1846#	1853#	1881#	1888#	1913#	1920#	1936#	1965#	1968#	1975#	1988#	2014#	2017#	
2024#	2036#	2060#	2063#	2070#	2082#	2111#	2114#	2121#	2133#	2160#	2163#	2170#	
2183#	2208#	2211#	2218#	2230#	2253#	2256#	2263#	2275#	2297#	2300#	2307#	2319#	
2342#	2345#	2352#	2392#	2399#	2427#	2434#	2459#	2466#	2514#	2521#	2569#	2576#	
2627#	2633#	2698#	2704#	2731#	2738#	2806#	2813#	2854#	2861#	2877#	2918#	2925#	
2932#	2951#	2969#	2974#	2980#	2999#	3017#	3022#	3028#	3048#	3074#	3079#	3086#	
3107#	3181#	3186#	3193#	3216#	3335#	3346#	3353#	3376#	3512#	3523#	3529#	3627#	
3633#	3730#	3737#	3760#	3896#	3906#	3912#	3937#	4071#	4078#	4082#	4084#	4114#	
4129#	4133#	4135#	4141	4153#	4160#								
17#	63	76#	192	196#	435	437#	440	444#	688	690#	703	705#	
717	719#	765	768#	887	889#	913	917#	927	929#	939	944#	1464	
1470#	1505	1512#	1548	1555#	1590	1597#	1632	1638#	1673	1680#	1716	1723#	
1758	1765#	1800	1806#	1846	1853#	1881	1888#	1913	1920#	1968	1975#	2017	
2024#	2063	2070#	2114	2121#	2163	2170#	2211	2218#	2256	2263#	2300	2307#	
2345	2352#	2392	2399#	2427	2434#	2459	2466#	2514	2521#	2569	2576#	2627	
2633#	2698	2704#	2731	2738#	2806	2813#	2854	2861#	2925	2932#	2974	2980#	
3022	3028#	3079	3086#	3186	3193#	3346	3353#	3523	3529#	3627	3633#	3730	
3737#	3906	3912#	4078	4082#	4129	4133#	4160						
446#	452	456#	461	465#	477	482#	503	509#	523	527#	532	536#	

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

	550	554#	567	571#	582	586#	599	692#	700	707#	714	770#	883
	891#	909	919#	923	931#	935	1401#	1405	1936#	1965	1988#	2014	2036#
	2060	2082#	2111	2133#	2160	2183#	2208	2230#	2253	2275#	2297	2319#	2342
	2877#	2918	2951#	2969	2999#	3017	3048#	3074	3107#	3181	3216#	3335	3376#
	3512	3760#	3896	3937#	4071	4084#	4114	4135#	4141	4153			
TSSAVL= 177777	8#												
TSSEGL= 177777	8#	1936#	1957	1965#	1967	1988#	2006	2014#	2016	2036#	2052	2060#	2062
	2082#	2103	2111#	2113	2133#	2154	2160#	2162	2183#	2202	2208#	2210	2230#
	2247	2253#	2255	2275#	2291	2297#	2299	2319#	2336	2342#	2344	2877#	2911
	2918#	2920	2951#	2969	2971	2999#	3017	3019	3048#	3074	3076	3107#	3181#
TSSEKO= 010000	3183	3216#	3335#	3337	3376#	3512#	3514	3760#	3896#	3898	3937#	4071#	4073
	1936#	1957	1965	1988#	2006	2014	2036#	2052	2060	2082#	2103	2111	2133#
	2154	2160	2183#	2202	2208	2230#	2247	2253	2275#	2291	2297	2319#	2336
	2342	2877#	2911	2918	2951#	2969	2999#	3017	3048#	3074	3107#	3181	3216#
TSUBN= 000000	3335	3376#	3512	3760#	3896	3937#	4071						
	8#	1470#	1512#	1555#	1597#	1638#	1680#	1723#	1765#	1806#	1853#	1888#	1920#
	1975#	2024#	2070#	2121#	2170#	2218#	2263#	2307#	2352#	2399#	2434#	2466#	2521#
	2576#	2633#	2704#	2738#	2813#	2861#	2932#	2980#	3028#	3086#	3193#	3353#	3529#
TSTAGL= 177777	3633#	3737#	3912#										
TSTAGN= 010074	8#												
	8#	446#	456#	465#	482#	509#	527#	536#	554#	571#	586#	692#	707#
	770#	891#	919#	931#	1401#	1470#	1512#	1555#	1597#	1638#	1680#	1723#	1765#
	1806#	1853#	1888#	1920#	1975#	2024#	2070#	2121#	2170#	2218#	2263#	2307#	2352#
	2399#	2434#	2466#	2521#	2576#	2633#	2704#	2738#	2813#	2861#	2932#	2980#	3028#
TSTEMP= 000000	3086#	3193#	3353#	3529#	3633#	3737#	3912#	4084#	4135#				
	63#	192#	435#	440#	452#	461#	477#	503#	523#	532#	550#	567#	582#
	599#	688#	700#	703#	714#	717#	722#	723#	724#	725#	726#	727#	728#
	729#	730#	731#	732#	733#	734#	735#	736#	737#	738#	739#	740#	741#
	742#	743#	744#	745#	746#	747#	748#	749#	750#	751#	752#	753#	754#
	755#	756#	757#	758#	759#	760#	761#	762#	763#	765#	883#	887#	909#
	913#	923#	927#	935#	939#	1405#	1464#	1471#	1477#	1505#	1515#	1521#	1548#
	1556#	1562#	1590#	1598#	1604#	1632#	1639#	1645#	1673#	1683#	1689#	1716#	1724#
	1730#	1758#	1766#	1772#	1800#	1808#	1818#	1846#	1855#	1860#	1881#	1890#	1895#
	1913#	1924#	1930#	1956#	1957#	1965#	1968#	1977#	1983#	2005#	2006#	2014#	2017#
	2026#	2031#	2051#	2052#	2060#	2063#	2071#	2077#	2102#	2103#	2111#	2114#	2123#
	2128#	2153#	2154#	2160#	2163#	2172#	2178#	2201#	2202#	2208#	2211#	2220#	2225#
	2246#	2247#	2253#	2256#	2265#	2270#	2290#	2291#	2297#	2300#	2309#	2314#	2335#
	2336#	2342#	2345#	2354#	2364#	2392#	2401#	2406#	2427#	2436#	2441#	2459#	2468#
	2475#	2514#	2522#	2529#	2569#	2579#	2586#	2627#	2636#	2642#	2698#	2708#	2714#
	2731#	2740#	2744#	2762#	2763#	2806#	2815#	2826#	2854#	2863#	2868#	2889#	2890
	2894#	2895	2910#	2911#	2918#	2925#	2934#	2939#	2969#	2974#	2982#	2987#	3017#
	3022#	3030#	3035#	3074#	3079#	3088#	3093#	3181#	3186#	3195#	3205#	3335#	3346#
	3355#	3365#	3512#	3523#	3531#	3537#	3627#	3635#	3641#	3730#	3739#	3749#	3896#
	3906#	3914#	3927#	4071#	4078#	4087#	4091#	4096#	4101#	4107#	4114#	4129#	4137#
	4143#	4149#	4153#	4160#									
TSTEST= 000051	8#												
	8#	1470#	1512#	1555#	1597#	1638#	1680#	1723#	1765#	1806#	1853#	1888#	1920#
	1975#	2024#	2070#	2121#	2170#	2218#	2263#	2307#	2352#	2399#	2434#	2466#	2521#
	2576#	2633#	2704#	2738#	2813#	2861#	2932#	2980#	3028#	3086#	3193#	3353#	3529#
TSSTM= 177777	3633#	3737#	3912#										
	8#	453	462	473	478	488	499	504	518	524	533	546	551
	561	568	578	583	595	600	610	623	633	644	655	664	671
	680	772	775	780	785	790	805	818	846	851	860	865	867
	873	877	884	894	901	906	910	924	936	946	959	964	966
	1049	1411	1423	1428	1443	1450	1486	1492	1498	1503	1506	1529	1535
	1541	1546	1549	1571	1577	1583	1588	1591	1613	1619	1625	1630	1633
	1654	1660	1666	1671	1674	1697	1703	1709	1714	1717	1739	1745	1751

F10

PAGE: 0122AS
SEQ 0122

CVRLAA.SUP

11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

1756	1759	1781	1787	1793	1798	1801	1823	1831	1840	1847	1869	1874
1882	1901	1906	1914	1936	1951	1956	1966	1969	1988	2000	2005	2015
2018	2036	2046	2051	2061	2064	2082	2097	2102	2112	2115	2133	2148
2153	2161	2164	2183	2196	2201	2209	2212	2230	2241	2246	2254	2257
2275	2285	2290	2298	2301	2319	2330	2335	2343	2346	2369	2377	2386
2393	2415	2420	2428	2447	2452	2460	2492	2497	2506	2515	2548	2553
2563	2570	2605	2610	2619	2628	2662	2667	2676	2681	2689	2699	2725
2732	2758	2762	2770	2776	2784	2790	2798	2807	2835	2842	2848	2855
2877	2882	2889	2894	2904	2910	2919	2926	2951	2958	2962	2967	2970
2975	2999	3006	3010	3015	3018	3023	3048	3051	3059	3062	3066	3072
3075	3080	3107	3110	3116	3120	3133	3137	3143	3145	3158	3164	3169
3172	3178	3182	3187	3216	3225	3228	3234	3239	3249	3254	3260	3265
3271	3276	3291	3306	3309	3321	3328	3333	3336	3347	3376	3381	3389
3392	3396	3402	3405	3411	3416	3426	3431	3437	3442	3448	3453	3468
3483	3486	3498	3505	3510	3513	3524	3556	3563	3566	3570	3576	3591
3606	3609	3621	3628	3660	3667	3670	3674	3680	3695	3710	3713	3725
3731	3760	3765	3773	3776	3780	3786	3789	3795	3800	3810	3815	3821
3826	3832	3837	3852	3867	3870	3882	3889	3894	3897	3907	3937	3940
3943	3951	3955	3961	3964	3970	3975	3985	3990	3996	4001	4007	4012
4027	4042	4045	4057	4064	4069	4072	4079					
1975#	1470#	1512#	1555#	1597#	1638#	1680#	1723#	1765#	1806#	1853#	1888#	1920#
2576#	2024#	2070#	2121#	2170#	2218#	2263#	2307#	2352#	2399#	2434#	2466#	2521#
3633#	2633#	2704#	2738#	2813#	2861#	2932#	2980#	3028#	3086#	3193#	3353#	3529#
931#	935											
891#	909											
919#	923											
4084#	4115											
692#	700											
770#	883											
446#	452	456#	461	465#	477	482#	503	509#	523	527#	532	536#
550	554#	567	571#	582	586#	599						
1936#	1956	1965#	1988#	2005	2014#	2036#	2051	2060#	2082#	2102	2111#	2133#
2153	2160#	2183#	2201	2208#	2230#	2246	2253#	2275#	2290	2297#	2319#	2335
2342#	2877#	2910	2918#	2951#	2969#	2999#	3017#	3048#	3074#	3107#	3181#	3216#
3335#	3376#	3512#	3760#	3896#	3937#	4071#						
4135#	4154											
1401#	1405											
707#	714											
1470#	1505	1512#	1548	1555#	1590	1597#	1632	1638#	1673	1680#	1716	1723#
1758	1765#	1800	1806#	1846	1853#	1881	1888#	1913	1920#	1968	1975#	2017
2024#	2063	2070#	2114	2121#	2163	2170#	2211	2218#	2256	2263#	2300	2307#
2345	2352#	2392	2399#	2427	2434#	2459	2466#	2514	2521#	2569	2576#	2627
2633#	2698	2704#	2731	2738#	2762	2806	2813#	2854	2861#	2889	2894	2925
2932#	2974	2980#	3022	3028#	3079	3086#	3186	3193#	3346	3353#	3523	3529#
3627	3633#	3730	3737#	3906	3912#	4078						
246#	825#	1864	1991	2187	2410	2717	2746	2829	2870	2940	2988	3037
3095	3206	3366	3538	3642	3750	3928						
711#	837											
722	1470#											
731	1853#											
732	1888#											
733	1920#											
734	1975#											
735	2024#											
736	2070#											

TSTSTS= 000001
 TSSAU = 010017
 TSSCLE= 010015
 TSSDU = 010016
 TSSHAR= 010072
 TSSHW = 010012
 TSSINI= 010014
 TSSMSG= 010011
 TSSSEG= 010000
 TSSSOF= 010073
 TSSSRV= 010020
 TSSSW = 010013
 TSSTES= 010071
 T.CNTL 002260
 T.SIZE 013334
 T1 016432 G
 T10 017512 G
 T11 017564 G
 T12 017622 G
 T13 017742 G
 T14 020044 G
 T15 020132 G

G10

PAGE: 0123AS
SEQ 0123

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CVRLAA.SUP 11-APR-78 09:27

CROSS REFERENCE TABLE -- USER SYMBOLS

XEQ.IN	036616	4175#																		
XEQ.LA	032706	4175#																		
XEQ.OP	036710	4175#																		
XEQ.PR	032110	4175#																		
XEQ.TE	036754	4175#																		
XPOLY	002202	223#	1370	1381																
XTIME	056276	4175#																		
XTIMEN	057122	4175#																		
XTINST	056320	4175#*																		
XXOP.D	034464	4175#																		
XSALWA=	000000	8#																		
XSFAIS=	000040	8#	4141																	
XSOFFS=	000400	8#	4141																	
XSTRUE=	000020	8#																		
SBREG	035160	4175#*																		
SENDAD	057376	4175#																		
SSAV2	060442	4175#																		
SSAV3	060456	4175#																		
SSAV4	060474	4175#																		
SSAV5	060514	4175#																		
.	= 060720	10#																		
		2247	68#	72#	429#	432#	433#	439#	1957	2006	2052	2103	2154	2202						
			2291	2336	2763	2830	2895	2911	4126#	4141	4157#	4161#	4175#							

. ABS. 060720 000

ERRORS DETECTED: 0

DSKZ:CVRLAA DSKZ:CVRLAA/CRF:SYM/SOL=DSKZ:CVRLAA.SML,DSKM:CVRLAA.P11,DSKZ:CVRLAA.SUP
 RUN-TIME: 50 54 4 SECONDS
 RUN-TIME RATIO: 1395/109=12.7
 CORE USED: 18K (35 PAGES)

