

MB061,RLV12

RLV12 DISKLESS
CVRLBA0

AH-S827A MC
FICHE 1 OF 1

JUL 1982
COPYRIGHT © 1982
MADE IN USA



Table with 12 columns and 12 rows. Each cell contains a small table or data structure, likely representing a grid of data points or a matrix. The content is too small to read accurately but appears to be organized in a structured format.

IDENTIFICATION

PRODUCT CODE: AC-S825A-MC
PRODUCT NAME: CVRLBA0 RLV12 DISK ESS
DATE CREATED: FEB 1982
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: G. PASQUANTONIO

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) 1982 DIGITAL EQUIPMENT CORPORATION

1.0 GENERAL INFORMATION

C 1

SEQ 0002

1.1 ABSTRACT

THE RLV12 DISKLESS TEST IS AN LSI-11 (PDP-11) BASED PROGRAM THAT WILL TEST THE RLV12, RLV11 AND/OR RL11 DISK CONTROLLERS WITH OR WITHOUT DRIVE ATTACHED. THE PROGRAM IS AN ADAPTATION OF "CVRLABO RLV11 RLO1 DSKLS" RETAINING ALL PREVIOUS TEST CAPABILITIES, AND UPGRADED TO INCLUDE ADDITIONAL TESTING IN SUPPORT OF THE RLV12.

RLV12 MODE

THE PROGRAM TESTS THE BASIC INTERFACE LOGIC, CONTROL REGISTER MANIPULATION, AND FUNCTIONALITY. THE RLV12 MAINTENANCE MODE FUNCTION IS EXECUTED TO TEST THE CONTROLLER WRITE/READ DATA PATHS WITHOUT A DRIVE PRESENT. THE EXTENDED ADDRESSING CAPABILITY IS TESTED IN 18 OR 22 BIT MODE DEPENDING ON THE TYPE AND AMOUNT OF MEMORY INSTALLED IN THE TEST SYSTEM. ADDITIONALLY, THE DRIVE BUS INTERFACE LOGIC IS TESTED PROVIDED THE G5388 TEST LOOP MODULE (TLM) IS INSTALLED.

NOTE: THE TLM IS AN IN-HOUSE SPECIAL TEST DEVICE DESIGNED FOR MANUFACTURING USE. IT PROBABLY WILL NOT BE GENERALLY AVAILABLE IN THE FIELD. THE CODE FOR TESTING WITH THE TLM IS INCLUDED IN THIS PROGRAM (BYPASSED BY DEFAULT) TO MAINTAIN CONSISTENCY WITH THE MANUFACTURING VERSION.

RLV11 MODE

SAME AS ABOVE EXCEPT THAT EXTENDED ADDRESSING IS LIMITED TO 18 BITS, AND DRIVE INTERFACE IS NOT TESTED.

RL11 MODE

SINCE THE MAINTENANCE FUNCTION DOES NOT EXIST ON THE RL11, THIS PROGRAM WILL ONLY TEST THE BASIC INTERFACE LOGIC, AND REGISTER FUNCTIONALITY. THE "NOP" COMMAND IS THE ONLY FUNCTION EXECUTED ON THE RL11.

THIS DIAGNOSTIC IS DESIGNED TO RUN UNDER XXDP+ AND REQUIRES THE SERVICES OF THE DIAGNOSTIC SUPERVISOR (DRS REV D). IT WILL RUN STANDALONE UNDER XXDP+ AND IS CHAINABLE UNDER XXDP+, ACT, OR APT.

1.2 HARDWARE REQUIREMENTS

LSI-11 (PDP-11) PROCESSOR WITH 16K OR MORE OF CORE.
MEMORY MANAGEMENT, K111 OR EQUIVALENT (OPTIONAL).
CONSOLE TERMINAL (LA30, LA36, VT52, VT100, ETC.).
XXDP+ LOAD DEVICE AND DIAGNOSTIC MEDIA (RX01, RX02, ETC.).
RLV12, RLV11, OR RL11 CONTROLLER(S) UNDER TEST (1-8).
G5388 RLV12 TEST LOOP MODULE (1-8, OPTIONAL).

1.3 RELATED DOCUMENTS AND STANDARDS

 CHQUSB XXDP+/SUPR USER MANUAL.
 RLV12 DISK CONTROLLER USERS GUIDE.
 G5388 TLM ENGINEERING SPEC (OPTIONAL).

1.4 MISCELLANEOUS

 ALL HARDWARE OTHER THAN THE UNIT(S) UNDER TEST IS ASSUMED TO
 BE IN PROPER WORKING ORDER. IF NOT (OR YOU DON'T KNOW) RUN
 ALL APPLICABLE SYSTEM DIAGNOSTICS.

2.0 LOADING AND STARTING PROCEDURES

 THIS PROGRAM IS LOADED AND STARTED FROM ANY XXDP+ MEDIA
 USING THE STANDARD XXDP+ OPERATING PROCEDURES.

AT START UP, THE SUPERVISOR WILL IDENTIFY ITSELF AND THE
 NAME OF THIS PROGRAM ON THE CONSOLE TERMINAL, AND THEN
 DISPLAY A COMMAND MODE PROMPT (DR>) WHICH INDICATES
 READY TO ACCEPT ANY OF THE COMMANDS DESCRIBED IN 2.1 BELOW.
 THE GENERALIZED COMMAND STRING FORMAT IS:

DR>COM(MAND)/SWITCH:VALUE/SWITCH:VALUE ... <CR>

2.1 SUPERVISOR COMMAND SUMMARY

 THIS SECTION PRESENTS A BRIEF OVERVIEW OF THE COMMANDS
 NECESSARY TO CONTROL THE OPERATION OF THIS PROGRAM
 UNDER THE XXDP+ DIAGNOSTIC SUPERVISOR (REV D).

THE PRIMARY COMMANDS ARE:

STA(RT)	INITIAL START, BUILD P-TABLES (SEE 2.4).
RES(TART)	RESTART USING EXISTING P-TABLES.
CON(TINUE)	CONTINUE AFTER <^C> OR ERROR HALT.
PRO(CEED)	PROCEED (FROM ERROR HALT ONLY).
EXI(T)	RETURN TO XXDP+ MONITOR.

THE FOLLOWING SWITCHES APPLY TO THE ABOVE:

/TEST:<TEST NUMBERS TO RUN>
 /FLAG:<SEE FLAG LIST BELOW>
 /EOP:<NUMBER OF PASSES 'TIL END-OF-PASS REPORT>
 /PASS:<NUMBER OF PASSES TO RUN>

ADDITIONAL COMMANDS AVAILABLE ARE:

DRO(P)/UNIT:N	REMOVE UNIT N FROM TEST LIST.
ADD/UNIT:N	ADD UNIT N (PREVIOUSLY DROPPED).
DIS(PLAY)/UNIT:N	PRINT UNITS P-TABLE ENTRIES.
PRI(NT)	PRINT OPTIONAL REPORTS (IF ANY).
FLA(GS)	PRINT CURRENT FLAG SETTINGS.
ZFL(AGS)	CLEAR ALL FLAGS.

2.2 RUN TIME OPTIONS (FLAGS)

 THE FOLLOWING FLAGS ARE USED IN LIEU OF THE HARDWARE
 SWITCH REGISTER TO FURTHER DEFINE PROGRAM BEHAVIOUR:

HOE	HALT ON ERROR
LOE	LOOP ON ERROR
IER	INHIBIT ALL ERROR REPORTS
IBR	INHIBIT BASIC ERROR REPORTS
IXR	INHIBIT EXTENDED ERROR REPORTS
PRI	DIRECT OUTPUT TO A LINE PRINTER
PNT	PRINT TEST NUMBERS AS EXECUTED
BOE	GOOD OLD "BELL-ON-ERROR"
UAM	RUN IN "UNATTENDED MODE" (UNUSED)
ISR	INHIBIT STATISTICAL REPORTS
IDR	INHIBIT "AUTO-DROP"
ADR	EXECUTE "AUTO-DROP" IF DEVICE NOT READY
LOT	LOOP ON TEST
EVL	EVALUATE ERRORS (UNUSED)

FLAG SETTINGS ARE ALTERED BY USING THE /FLAGS: SWITCH
 IN ANY COMMAND STRING. TWO EXAMPLES FOLLOW:

STA/FLA:IER:BOE<CR>
 START THE PROGRAM, RUN ALL TEST IN ORDER, INHIBIT ERROR
 REPORTS AND RING BELL INSTEAD.

RES/TES:1-5,7,9-29/PAS:10/FLA:HOE:BOE<CR>
 RESTART AND RUN ALL TESTS EXCEPT 6, 8, AND 30 FOR 10 PASSES.
 RING THE BELL AND HALT (RETURN TO COMMAND MODE) ON ANY ERROR.

2.3 CONTROL CHARACTERS

 THE FOLLOWING CONTROL CHARACTERS ARE RECOGNIZED:

CTRL C <^C>	ABORT AND RETURN TO COMMAND MODE.
CTRL O <^O>	SUPPRESS TTY OUTPUT UNTIL NEXT <^O>.
CTRL Z <^Z>	DURING START/RESTART PARAMETER ENTRY ONLY. IMPLIES TAKE DEFAULTS ON ALL REMAINING QUERIES.

HARD AND SOFT PARAMETER SETUP

 THE FOLLOWING DIALOGUE IS TAKEN AT START TIME TO ESTABLISH THE
 HARDWARE AND SOFTWARE PARAMETER TABLES (P-TABLES) REQUIRED BY
 THE PROGRAM. DEFAULT VALUES, WHEN APPLICABLE, APPEAR JUST TO
 THE LEFT OF THE QUESTION MARK (?). TYPE <CR> TO ACCEPT THE
 DEFAULT, OR ANSWER THE QUERY APPROPRIATELY. A <^Z> TYPED AT
 ANY POINT DURING THE SESSION IMPLIES TAKE DEFAULTS FOR ALL
 REMAINING QUERIES. NOTE THAT ON RESTART OR CONTINUE, ONLY
 THE SOFTWARE DIALOGUE IS TAKEN.

QUERY -----	RESPONSE -----
CHANGE HW (L) ?	NO DEFAULT, ANSWER Y OR N. IF N, ASSUME 1 UNIT IN THE DEFAULT CONFIGURATION, AND GO TO 'CHANGE SW'.
# UNITS (D) ?	NO DEFAULT, ENTER # OF UNITS TO TEST.
RLV12 (L) Y ?	ANSWER Y OR N. IF N, SKIP NEXT QUERY.
BAE ENABLED (L) Y ?	ON RLV12, BAE IS ENABLED BY DEFAULT... ...ANSWER Y OR N, AND SKIP NEXT QUERY.
RLV11 (L) N ?	ANSWER Y IF RLV11, N IF RL11.
CSR ADDRESS (O) 174400 ?	ENTER RL CSR ADDRESS IF DIFFERENT.
VECTOR (O) 160 ?	ENTER INTERRUPT VECTOR IF DIFFERENT.
BR LEVEL (O) 4 ?	ENTER INTERRUPT PRIORITY IF DIFFERENT.
CHANGE SW (L) ?	NO DEFAULT, ANSWER Y OR N. IF N, BYPASS ALL REMAINING QUERIES.
ERROR LIMIT FOR AUTO-DROP (O) 0 ? SET ERROR LIMIT (IF ANY).	
ALL REMAINING QUERIES ARE FOR OPTIONAL (MANUFACTURING) G5388 TEST-LOOP-MODULE SET-UP. USE <^Z> TO BYPASS.	
G5388 TLM INSTALLED (L) N ?	ANSWER Y OR N. IF NO, BYPASS THE REST.
CSR ADDRESS (UNIT 0) (O) 160010	? BASE TLM CSR (SEE NOTE BELOW).
DUMP PROM ON TTY: (L) N ?	IF Y, GO DIRECTLY TO THE DUMP... ...UTILITY (SEE 7.16.7 BELOW).
PROM ID NUMBER (O) 401 ?	ENTER PROM ID IF DIFFERENT.
INHIBIT MAX PEAK SHIFT (L) N ?	ANSWER Y OR N...
INHIBIT MIN PEAK SHIFT (L) N ?	
INHIBIT NOMINAL CLOCK (L) N ?	
INHIBIT FAST CLOCK (L) N ?	
INHIBIT SLOW CLOCK (L) N ?	
SECTOR NUMBER (O) 0 ?	SECTOR NUMBER TO TEST (0 = ALL).

NOTE: IF RUNNING MULTIPLE UNITS WITH TLM'S (UNLIKELY), THE PROGRAM
 EXPECTS THAT EACH ADDITIONAL TLM CSR IS OFFSET BY 10 (OCTAL).
 I.E. 160010, 160020, 160030, ETC...

EXECUTION TIME

 EXECUTION TIME IS DEPENDANT UPON CPU TYPE, MEMORY SIZE, AND
 BUS TYPE (18 VS 22 BIT), BUT SHOULD NOT EXCEED 30 SECONDS IN
 ANY CONFIGURATION.

3.0 ERROR REPORTING

ALL ERRORS ARE REPORTED ON THE CONSOLE TERMINAL AS THEY OCCUR.
THE GENERAL ERROR FORMAT IS:

CVRLB XXX ERR NNNNN UNIT NN TEST TTT SUB SSS PC: PPPPP
ONE LINE DESCRIPTION.....
EXP'D: 000000 REC'D: 177777

WHERE: XXX IS ERROR TYPE (HRD OR SFT),
NNNNN IS THE ERROR NUMBER,
NN IS THE FAILING UNIT NUMBER,
TTT IS THE TEST,
SSS AND SUBTEST NUMBERS, AND
PPPPP IS THE PC OF THE ERROR CALL.

IN MANY CASES, THE ENTIRE CONTROLLER STATE (ALL REGISTERS
BEFORE AND AFTER THE FACT) IS ALSO DISPLAYED.

ERROR REPORTING AND RECOVERY MAY BE ALTERED AND/OR INHIBITED
VIA THE /FLAG: SWITCH AS DESCRIBED IN 2.2 ABOVE.

4.0 PERFORMANCE AND PROGRESS REPORTS

THE OPERATING ENVIRONMENT IS DISPLAYED ON THE CONSOLE
TERMINAL AT START/RESTART TIME.

A TOTAL (CUMULATIVE) ERROR COUNT IS DISPLAYED AT THE END
OF EVERY PASS, THEREAFTER.

5.0 RLV12 CONTROLLER REGISTER DEFINITION

THE RLV12, RLV11, AND RL11 CONTROLLERS UTILIZE THE FOLLOWING
REGISTERS FOR CONTROL OF THE SUBSYSTEM:

5.1 RLCS -- CONTROL AND STATUS REGISTER (174400)

BIT<15> COMPOSITE ERROR
BIT<14> DRIVE ERROR
BIT<13> NON-EX MEM (OPI=0), OR PARITY (OPI=1)
BIT<12> DATA LATE (OPI=0), OR HEADER NOT FOUND (OPI=1)
BIT<11> DATA CRC (OPI=0), OR HEADER CRC (OPI=1)
BIT<10> OPERATION INCOMPLETE (OPI)
BIT<9:8> DRIVE SELECT (3 TO 0)
BIT<7> CONTROLLER READY (OR DONE)
BIT<6> INTERRUPT ENABLE
BIT<5:4> EXTENDED ADDRESS BITS<17:16>
BIT<3:1> FUNCTION CODE:
0 = MAINT (RLV12/11), NOP (RL11)
1 = WRITE CHECK
2 = GET DRIVE STATUS
3 = SEEK
4 = READ HEADER
5 = WRITE DATA
6 = READ DATA
7 = READ WITHOUT HEADER CHECK
BIT<0> DRIVE READY

- 5.2 RLBA -- BUS ADDRESS REGISTER (174402)

BIT<15:0> BUS ADDRESS FOR DMA DATA EXCHANGE.
- 5.3 RLDA -- DISK ADDRESS REGISTER (174404)

FOR READ/WRITE FUNCTIONS:
BIT<15:7> CYLINDER NUMBER
BIT<6> HEAD SELECT
BIT<5:0> SECTOR NUMBER
- FOR SEEK/STATUS FUNCTIONS:
BIT<15:7> SEEK DIFFERENCE (SEEK), UNUSED (STATUS)
BIT<6:5> ZERO
BIT<4> HEAD SELECT (SEEK), ZERO (STATUS).
BIT<3> ZERO (SEEK), RESET (STATUS)
BIT<2> SEEK DIRECTION (SEEK), ZERO (STATUS)
BIT<1> 0 = SEEK, 1 = GET STATUS
BIT<0> MARKER, MUST BE 1 FOR EITHER FUNCTION.
- 5.4 RLMP -- MULTIPURPOSE REGISTER (174406)

WRITE BEFORE READ/WRITE DATA FUNCTIONS:
BIT<15:0> NEGATIVE WORD COUNT
- READ AFTER MAINTENANCE FUNCTION:
1ST WORD: CRC OF INITIAL (DA)+3
2ND WORD: CRC OF CRC OF INITIAL (DA)+4
- READ AFTER READ-HEADER FUNCTION:
1ST WORD: DISK ADDRESS.
2ND WORD: ZERO.
3RD WORD: HEADER CRC.
- READ AFTER GET STATUS FUNCTION:
BIT<15> WRITE DATA ERROR (WDE)
BIT<14> CURRENT HEAD ERROR (CHE)
BIT<13> WRITE LOCK (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> ZERO
BIT<6> HEAD SELECT (HS)
BIT<5> COVER OPEN (CO)
BIT<4> HEADS OUT (HO)
BIT<3> BRUSHES HOME (BH)
BIT<2:0> DRIVE STATE:
0 = LOAD
1 = SPIN UP
2 = BRUSH CYCLE
3 = LOAD HEADS
4 = SEEK - TRACK COUNTING
5 = SEEK - LINEAR MODE
6 = UNLOAD HEADS
7 = SPIN DOWN

5.5 RLBAE -- BUS ADDRESS EXTENSION (174410) -- RLV12 ONLY

 BIT<15:6> ZERO (UNUSED)
 BIT<5:0> EXTENDED ADDRESS BITS<21:16>

6.0 G5388 TLM REGISTER DEFINITION

 THE G5388 TLM UTILIZES THE FOLLOWING REGISTERS:

6.1 TCSR -- CONTROL AND STATUS (160010)

 BIT<15> NEW SEEK/STATUS COMMAND RECEIVED.
 BIT<14> WRITE GATE ERROR.
 BIT<13> WRITE GATE.
 BIT<12> WRITE DATA (LATCH).
 BIT<11> SYS-CLK.
 BIT<10:9> DRIVE NUMBER SELECTED.
 BIT<8> PWR-OK.
 BIT<7> DRIVE READY.
 BIT<6> DRIVE ERROR.
 BIT<5> PEAK SHIFT ENABLE.
 BIT<4> SECTOR GENERATOR ENABLE.
 BIT<3:2> CLOCK SELECT:
 0 - SLOW
 1 - NOMINAL
 2 - FAST
 3 - PROM READ MODE
 BIT<1> CLEAR FLAGS.
 BIT<0> TLM RESET.

6.2 TSKGS -- SEEK/STATUS REGISTER (160012)

 BIT<15:0> HOLDS DRIVE COMMAND RECEIVED FROM RLV12.

6.3 TPDS -- PSUEDO-DRIVE STATUS REGISTER (160014)

 BIT<15:0> SENDS PSUEDO-DRIVE STATUS TO THE RLV12.

6.4 TPROM -- PROM REGISTER (160016)

 BIT<15:0> HOLDS PROM ID DURING 'TLM RESET' AND/OR
 PROM DATA DURING 'READ PROM' MODE.

7.0 HARDWARE TEST DESCRIPTION

THIS SECTION PROVIDES A BRIEF DESCRIPTION OF THE HARDWARE TESTS.
REFER TO THE PROGRAM LISTING (8.0) FOR FURTHER DETAILS.

7.1 TESTS 1 THRU 5 -- REGISTER ADDRESSABILITY.

THESE TESTS VERIFY THAT EACH CONTROLLER REGISTER ANSWERS TO
IT'S BUS ADDRESS. BOTH READ AND WRITE ACCESS IS VERIFIED
USING 'MOV' AND 'TST' INSTRUCTIONS.

7.2 TEST 6 -- BUS RESET.

VERIFIES THAT A 'BUS-RESET' PLACES THE CONTROLLER IF THE
PROPER 'INITIALIZED' STATE.

7.3 TESTS 7 THRU 10 -- READ AND WRITE REGISTERS.

VERIFIES THAT WE CAN WRITE TO AND READ FROM ALL REGISTERS
(EXCEPT MPR). VARIOUS DATA PATTERNS ARE WRITTEN AND VERIFIED
USING 'MOV' INSTRUCTIONS.

7.4 TESTS 11 THRU 14 -- BIS AND BIC REGISTERS.

VERIFIES THAT WE CAN SET AND CLEAR ALL WRITEABLE BITS IN ALL
REGISTERS (EXCEPT MPR) USING 'BIS' AND 'BIC' INSTRUCTIONS.
VARIOUS DATA PATTERNS ARE EMPLOYED HERE TOO.

7.5 TESTS 15 THRU 19 -- REGISTER UNIQUENESS.

VERIFIES THAT WRITING TO ANY REGISTER HAS NO AFFECT ON ANY
OTHER (NO DUAL-ADDRESSING).

EXCEPTION: EXTENDED ADDRESS BITS<17:16> MAY BE WRITTEN VIA
EITHER RLCS<5:4> OR RLBAE<1:0>. REGARDLESS OF WHICH REGISTER IS
ACTUALLY WRITTEN, THOSE BITS ARE COPIED INTO THE OTHER REGISTER.

7.6 TEST 20 -- FUNCTION CODE 0, MAINTENANCE OR NOP.

VERIFIES THAT FUNCTION CODE 0 EXECUTES AS A 'MAINTENANCE'
FUNCTION (RLV12 AND RLV11), OR 'NOP' (RL11).

FOR RLV12/11:
EXPECT TO EXECUTE A MAINTENANCE MODE SEQUENCE.
CHECK THAT THE MAINTENANCE FUNCTION COMPLETES WITH NO
CONTROLLER ERRORS, AND THAT THE FINAL VALUE IN THE RLDA
IS CORRECT, INDICATING THAT THE INTERNAL MAINTENANCE
MODE SEQUENCER FINISHED IT'S CYCLE.

FOR RL11:
EXPECT TO EXECUTE A 'NOP' FUNCTION.
CHECK THAT ALL ERROR BITS ARE CLEAR (RLCS<13:10>)
AND ALL OTHER REGISTERS UNAFFECTED.

- 7.7 TEST 21 -- INTERRUPT ON FUNCTION COMPLETE.

THIS TEST EXECUTES A NOP/MAINTENANCE FUNCTION WITH INTERRUPT ENABLED, AND VERIFIES THAT AN INTERRUPT THRU THE RL VECTOR ACTUALLY OCCURS. INCORRECT VECTORS UNDER 1000 ARE TRAPPED BY THE SUPERVISOR. INCORRECT VECTORS TO ANY OTHER ADDRESS WILL YIELD UNPREDICTABLE RESULTS.
- 7.8 TEST 22 -- INTERRUPT PRIORITY LEVEL TEST.

SIMILAR TO THE LAST TEST, EXCEPT THAT WE REITERATE AT ALL CPU LEVELS (7 TO 0). VERIFYING THAT THE INTERRUPT IS HELD OFF WHEN THE CPU LEVEL => CONTROLLER LEVEL.

NOTE: IF UNIT UNDER TEST IS RL11, YOU'LL SEE AN END-PASS AT THE COMPLETION OF THIS TEST. ALL FOLLOWING TESTS UTILIZE THE RLV12/11 MAINTENANCE FUNCTION.
- 7.9 TEST 23 -- MAINTENANCE, FORCE OPI ERROR (RLV12/11 ONLY).

THIS TEST WILL VERIFY THAT THE CONTROLLER WILL SET THE OPI (OPERATION INCOMPLETE) BIT ON A MAINTENANCE FUNCTION IF THE INITIAL WORD COUNT IS INCORRECT. WE'LL DO 2 PASSES.
1. WORD COUNT LESS THAN 510.
2. WORD COUNT GREATER THAN 511.
BOTH CASES SHOULD CAUSE AN OPI ERROR IN THE CONTROLLER.
- 7.10 TEST 24 -- MAINTENANCE, FORCE OPI AND INTERRUPT (RLV12/11 ONLY).

SIMILAR TO THE LAST, EXCEPT EXECUTE WITH INTERRUPT ENABLED. EXPECT OPI ERROR TO CAUSE AN INTERRUPT THRU THE RL VECTOR.
- 7.11 TEST 25 -- MAINTENANCE, OPI TIMING TEST (RLV12/11 ON LSI ONLY).

VERIFY THAT THE SETTING OF THE OPI BIT IS TIMED CORRECTLY. EXECUTE A MAINTENANCE FUNCTION IN FLAG MODE WITH INVALID WORD COUNT TO FORCE OPI ERROR. MEASURE THE TIME FROM "GO" UNTIL THE OPI ERROR FLAG SETS AND CHECK THAT TIME AGAINST THE SPEC LIMITS (155 TO 650 MSEC.).

NOTE: THIS TEST EMPLOYS A SOFT TIMER THAT HAS BEEN CALIBRATED FOR LSI11/2 AND /23 CPUS ONLY. IF YOUR CPU IS NOT ONE OF THESE THIS TEST IS AUTOMATICALLY BYPASSED.
- 7.12 TEST 26 -- MAINTENANCE, FIFO AND DMA TRANSFER (RLV12/11 ONLY).

VERIFIES THAT 'MAINTENANCE' FUNCTIONS CORRECTLY. EXECUTE A MAINTENANCE MODE FUNCTION IN FLAG MODE. VERIFY THE FINAL BA, DA, AND BAE REGISTERS ARE CORRECT. CHECK THE SERIAL WRITE/READ DATA PATHS BY READING THE TWO MAINTENANCE CRC WORDS (CRC OF DA+3, AND CRC OF CRC OF DA+4) FROM THE FIFO VIA THE MP REGISTER. VERIFY THAT 256 WORDS WERE TRANSFERRED FROM MEMORY TO THE FIFO, AND 255 WORDS FROM FIFO BACK TO MEMORY. REPEAT 57 TIMES USING VARIOUS DATA.
1. 28 DIFFERENT DATA PATTERNS.
2. THE COMPLIMENT OF THOSE PATTERNS.
3. ONE RANDOM 256 WORD PATTERN.

- 7.13 TEST 27 -- MAINTENANCE, FIFO ADDRESS TEST (RLV12/11 ONLY). L 1

 SIMILAR TO THE LAST, EXCEPT THAT THE DATA IS AN ADDRESS PATTERN WHERE EACH FIFO LOCATION WILL BE WRITTEN WITH IT'S OWN ADDRESS (0 TO 255.). REPEAT A SECOND TIME WITH A COMPLIMENT ADDRESS PATTERN (-1 TO -256.).
 INTERRUPT ON DONE IS ENABLED AND EXPECTED DURING THIS TEST.
- 7.14 TEST 28 -- MAINTENANCE, BANK SELECT 7 AND NXM (RLV12/ ONLY).

 FOR RLV12:
 VERIFIES THAT BBS7 WILL SELECT THE I/O PAGE AND THAT ACCESS TO LOCATION 0 IN THAT PAGE WILL GENERATE NXM AND OPI ERRORS.
 NOTE: IF BANK 7 IS NOT PROPERLY SELECTED, 1000 BYTES STARTING AT XXXX1000 WILL PROBABLY GET CRUNCHED !!!
- FOR RLV11:
 SINCE RLV1 DOESN'T ASSERT BBS7, WE'LL EXECUTE THIS TEST ONLY IF MEMORY SIZE IS LESS THAN 124K WORDS.
- 7.15 TEST 29 -- MAINTENANCE, EXTENDED MEMORY ACCESS (RLV12/11 ONLY).

 THIS TEST WILL VERIFY THAT THE CONTROLLER CAN ADDRESS EXTENDED MEMORY UP TO 124K (18 BIT) OR 2044K (22 BIT) DEPENDING UPON CONTROLLER TYPE AND CONFIGURATION.
 IT REQUIRES THE SERVICES OF THE MEMORY MANAGEMENT FACILITY (KT11 OR EQUIVALENT IN 11/23) TO ESTABLISH AND VERIFY DATA BUFFERS IN MEMORY FROM 32K TO 2044K, IN 4K INCREMENTS.
1. SETUP -- USING MMU, INITIALIZE A WRITE BUFFER WITH A RANDOM DATA PATTERN, AND CLEAR THE READ BUFFER.
 2. EXECUTE -- RLV12/11 MAINTENANCE MODE.
 WRITE BUFFER => FIFO => READ BUFFER.
 3. VERIFY -- USING MMU, VERIFY THAT THE READ BUFFER RECEIVED A COPY OF THE DATA IN THE WRITE BUFFER.
 4. INCREMENT BUFFER ADDRESS BY 4K, AND REPEAT.
- NOTE THAT THE ENTIRE 18 (OR 22) BIT ADDRESS SPACE IS TESTED ON 4K BOUNDARIES, WHETHER MEMORY EXISTS THERE OR NOT.
 IF MEMORY EXISTS, VERIFY THE DATA EXCHANGE AS DESCRIBED.
 IF NOT, VERIFY THAT THE CONTROLLER GOT A 'NXM ERROR'.
 CONTINUE ALL THE ABOVE UNTIL THE I/O PAGE IS REACHED AT 760000 (18 BIT) OR 17760000 (22 BIT).
- OF COURSE, IF AN MMU ISN'T AVAILABLE, THEN ALL THIS IS ACADEMIC, AND WE'LL JUST FALL THRU TO THE NEXT TEST.

7.16 DRIVE INTERFACE TESTS (RLV12 ONLY, G5388 TLM REQ'D).

THIS IS AN ASSORTMENT OF 6 TESTS AND 1 UTILITY DESIGNED TO VERIFY THE INTEGRITY OF THE DRIVE INTERFACE SECTION OF THE RLV12 CONTROLLER. ALL REQUIRE THE SERVICES OF THE SPECIAL TEST-LOOP-MODULE (TLM) DESIGNATED G5388.

FROM THE CONTROLLERS POINT OF VIEW, THE TLM LOOKS LIKE A 6 SECTOR RLO1/RLO2 DISK. VARIOUS HEADER AND DATA PATTERNS ARE STORED IN PROM ON THE TLM, AND ARE ACCESSABLE THRU THE EXECUTION OF 'NORMAL' READ/WRITE AND STATUS FUNCTIONS IN THE RLV12 CONTROLLER. REFER TO THE G5388 TLM ENGINEERING SPEC FOR FURTHER DETAILS.

7.16.1 TEST 30 -- STATIC CONTROL AND STATUS BITS.

CHECK THE STATIC CONTROL AND STATUS BITS TO AND FROM THE RLV12. THESE BITS INCLUDE SYS-CLK, PWR-OK, DRIVE-ERROR, DRIVE-READY, AND DRIVE-SELECTS.

7.16.2 TEST 31 -- DRIVE COMMAND, STATUS.

VERIFY THE DRIVE COMMAND AND STATUS CLOCK LOGIC BY EXECUTING A GET-STATUS COMMAND IN THE RLV12. CHECK THAT A DUMMY STATUS WORD IS CORRECTLY RECEIVED FROM THE TLM.

7.16.3 TEST 32 -- DRIVE COMMAND, SEEK.

VERIFY THE DRIVE COMMAND AND SECTOR PULSE LOGIC BY EXECUTING A SEEK COMMAND IN THE RLV12. CHECK THAT THE SEEK DIFFERENCE WORD WAS CORRECTLY TRANSMITTED TO THE TLM.

7.16.4 TEST 33 -- WRITE DATA PATH.

VERIFY THE WRITE GATE, WRITE GATE ERROR, AND WRITE DATA PATH BY EXECUTING A WRITE-DATA FUNCTION IN THE RLV12. THIS TEST ONLY PROVES THAT TRANSITIONS ARE TAKEING PLACE ON THE WRITE DATA LINE SINCE THE PSUEDO-SECTORS ARE ACTUALLY READ-ONLY (PROM).

7.16.5 TEST 34 -- READ DATA PATHS.

VERIFY THE READ DATA PATHS BY EXECUTING READ-DATA, READ-HEADER, AND READ-DATA-WITHOUT-HEADER FUNCTIONS IN THE RLV12. BY DEFAULT, EACH PSUEDO-SECTOR IS EXERCISED SIX TIMES, USEING THE FOLLOWING CLOCK VARIATIONS:

1. NOMINAL CLOCK, MIN PEAK SHIFT.
2. NOMINAL CLOCK, MAX PEAK SHIFT.
3. FAST CLOCK, MIN PEAK SHIFT.
4. FAST CLOCK, MAX PEAK SHIFT.
5. SLOW CLOCK, MIN PEAK SHIFT.
6. SLOW CLOCK, MAX PEAK SHIFT.

RETURNED DATA AND STATUS ARE VERIFIED IN EVEFY CASE. PEAK SHIFT AND CLOCKS MAY BE SELECTIVELY INHIBITED AND A SINGLE SECTOR EXERCISED (VS ALL OF THEM) AS DESCRIBED IN THE P-TABLE SOFT PARAMETER SETUP (SECTION 2.4) ABOVE.

7.16.6 TEST 35 -- WRITE CHECK.

VERIFY THE READ/WRITE DATA PATHS BY EXECUTING A WRITE-CHECK FUNCTION IN THE RLV12. EXECUTE USING CLOCK AND SECTOR OPTIONS AS DESCRIBED FOR TEST 34 ABOVE.

7.16.7 TLM PROM DUMP UTILITY.

THE FINAL SECTION IS A 'PROM DUMP UTILITY' ROUTINE WHICH CAN BE USED TO PRINT THE CONTENTS OF ANY OF THE 6 TLM PS 'EDO-SECTORS ON THE CONSOLE TERMINAL. THIS ROUTINE IS NOT INCLUDED IN THE NORMAL TEST SEQUENCE. IT MUST BE CALLED DIRECTLY AT START/RESTART TIME VIA THE SOFTWARE CHANGE QUERY 'DUMP TLM PROM ON TTY: ?'

WHEN CALLED, THE PROGRAM WILL INTRODUCE ITSELF, AND ASK FOR A TLM UNIT NUMBER (0 - 7), SECTOR NUMBER (1 - 6), AND DUMP FORMAT (OCTAL OR HEX). IT WILL THEN READ THAT SECTOR AND PRINT IT IN THE SPECIFIED FORMAT, 8 WORDS PER LINE FOR 20 LINES (160 WORDS TOTAL). WHEN YOU'RE DONE, TYPE <^C> TO EXIT BACK TO SUPERVISOR COMMAND MODE.

8.0 PROGRAM LISTING

THE PROGRAM LISTING FOLLOWS:

TABLE OF CONTENTS

14	PROGRAM HEADER
51	HARDWARE PARAMETER CODING
93	SOFTWARE PARAMETER CODING
145	GLOBAL EQUATES
221	GLOBAL DATA
452	INITIALIZATION CODE
630	GLOBAL SUBROUTINES
986	MEMORY SIZER
1094	REPORT ENVIRONMENT
1115	
1116	RL DISKLESS CONTROLLER TESTS.
1117	
1118	1 -- RLCS ADDRESSABILITY.
1140	2 -- RLBA ADDRESSABILITY.
1162	3 -- RLDA ADDRESSABILITY.
1184	4 -- RLMP ADDRESSABILITY.
1206	5 -- RLBAE ADDRESSABILITY (RLV12 ONLY).
1235	6 -- BUS RESET OF ALL REGISTERS.
1284	7 -- READ WRITE OF RLCS.
1313	8 -- READ WRITE OF RLBA.
1340	9 -- READ WRITE OF RLDA.
1363	10 -- READ WRITE OF RLBAE (RLV12 ONLY).
1390	11 -- BIS AND BIC OF RLCS.
1433	12 -- BIS AND BIC OF RLBA.
1470	13 -- BIS AND BIC OF RLDA.
1503	14 -- BIS AND BIC OF RLBAE (RLV12 ONLY).
1541	15 -- UNIQUENESS OF RLCS.
1580	16 -- UNIQUENESS OF RLBA.
1621	17 -- UNIQUENESS OF RLDA.
1662	18 -- UNIQUENESS OF RLMP.
1710	19 -- UNIQUENESS OF RLBAE (RLV12 ONLY).
1746	20 -- FUNCTION CODE 0, NOP (RL11), OR MAINT (RLV11/12).
1800	21 -- TEST INTERRUPT ON FUNCTION (0) COMPLETE.
1825	22 -- TEST INTERRUPT PRIORITY BR LEVEL.
1863	23 -- RLV11/12 MAINTENANCE, FORCED OPI (WC <> 511.)
1901	24 -- RLV11/12 MAINTENANCE, FORCED OPI INTERRUPT.
1923	25 -- RLV11/12 MAINTENANCE, OPI TIMING TEST.
1959	26 -- RLV11/12 MAINTENANCE, FIFO DMA AND CRC CHECK.
2092	27 -- RLV11/12 MAINTENANCE, FIFO ADDRESSING.
2184	28 -- RLV11/12 MAINTENANCE, BANK 7 SELECT AND NEXM TEST.
2230	29 -- RLV11/12 MAINTENANCE, EXTENDED MEMORY ACCESS TEST.
2355	
2356	RLV12 DRIVE INTERFACE TESTS (G5388 TLM REQUIRED).
2357	
2361	30 -- SYS CLK, PWR OK, DRIVE SELECT, READY, AND ERROR BITS.
2422	31 -- DRIVE COMMAND, STATUS AND STATUS CLOCK.
2459	32 -- DRIVE COMMAND, SEEK DIFF AND SECTOR PULSE.
2490	33 -- WRITE GATE, WRITE GATE ERROR, AND WRITE DATA.
2537	34 -- READ DATA, READ HEADER, AND READ DATA W/O HEADER.
2666	35 -- WRITE CHECK.
2864	TLM PROM DUMP UTILITY.

CVRLBAO -- RLV12 DISKLESS, MACY11 30G(1063) 08-FEB-82 10:07 C 2
CVRLBA.P11 08-FEB-82 09:59 TABLE OF CONTENTS

SEQ 0015

3071
3072 SUPERVISOR DISPATCH TABLE.
3092 GLOBAL ERROR HANDLERS AND ASCII TEXT
3343 DEVELOPMENT/DEBUG AIDS
3344
3345 RLV12 EMULATOR

```
14      :SBTTL PROGRAM HEADER
15      :XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
16      :XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
17      000000'      .MCALL SVC      : INITIALIZE SUPERVISOR MACROS
18      SVC
19      000001      SVCGBL= 1      : LIST GLOBAL TAGS AT RIGHT MARGIN.
20      000001      SVCTST= 1      : DITTO TEST TAGS.
21      000001      SVCSUB= 1      : DITTO SUBTEST TAGS.
22      000001      SVCTAG= 1      : DITTO ANY OTHER TAGS.
23      000001      SVCINS= 1      : DITTO INSTRUCTIONS AND DATA.
24      :
25      : THESE SYMBOLS CONTROL THE LISTING FIELD OF ALL SVC MACRO
26      : EXPANSIONS. YOU MAY CHANGE THEM AT ANY TIME OR PLACE.
27      :
28      : 1 = RIGHT-JUSTIFY (MAKES IT EASY TO DISTINGUISH
29      : SVC'S MACRO CODE FROM YOUR OWN).
30      : 0 = LEFT-JUSTIFY (ALIGN IN A NORMAL FASHION).
31      : -1 = DON'T LIST THE EXPANSIONS AT ALL.
32      :XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
33      :XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
34      002000      .ENABL ABS,AMA
35      002000      .= 2000
36      :
37      000100      PRGSIZ= ^H<LSLAST>      : PROGRAM SIZE IN 1/8 K UNITS (OCTAL).
38      000000      SVCGBL= 0      : ALIGN THE HEADER STUFF.
39      000000      SVCINS= 0
40      :
41      002000      POINTER BGNSFT,BGNSW,BGNDU,BGNAU,BGNRPT,BGNSETUP
42      002000      HEADER CVRLB,A,0,30,0,340
43      (4) 002000      LSNAME::      :DIAGNOSTIC NAME
44      (4) 002000      103      .ASCII /C/
45      (4) 002001      126      .ASCII /V/
46      (4) 002002      122      .ASCII /R/
47      (4) 002003      114      .ASCII /L/
48      (4) 002004      102      .ASCII /B/
49      (6) 002005      000      .BYTE 0
50      (6) 002006      000      .BYTE 0
51      (5) 002007      000      .BYTE 0
52      (5) 002010      LSREV::      :REVISION LEVEL
53      (4) 002010      101      .ASCII /A/
54      (5) 002011      LSDEPO::      :0
55      (4) 002011      060      .ASCII /O/
56      (5) 002012      LSUNIT::      :NUMBER OF UNITS
57      (4) 002012      000001      .WORD TSPTHV
58      (5) 002014      LSTIML::      :LONGEST TEST TIME
59      (4) 002014      000036      .WORD 30.
60      (5) 002016      LSHPCP::      :POINTER TO H.W. QUES.
61      (4) 002016      002172      .WORD LSHARD
62      (5) 002020      LSSPCP::      :POINTER TO S.W. QUES.
63      (4) 002020      002412      .WORD LSSOFT
64      (5) 002022      LSHPTP::      :PTR. TO DEF. H.W. PTABLE
65      (4) 002022      002376      .WORD LSHW
66      (5) 002024      LSSPTP::      :PTR. TO S.W. PTABLE
67      (4) 002024      003304      .WORD LSSW
68      (5) 002026      LSLADP::      :DIAG. END ADDRESS
69      (4) 002026      040344      .WORD LSLAST
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 E 2
PROGRAM HEADER PAGE 2-1

SEQ 0017

(5)	002030		LSSTA::		;RESERVED FOR APT STATS
(4)	002030	000000	LSSTA::	.WORD 0	
(5)	002032		LSCO::		
(4)	002032	000000	LSCO::	.WORD 0	
(5)	002034		LSDTYP::		;DIAGNOSTIC TYPE
(4)	002034	000000	LSDTYP::	.WORD 0	
(5)	002036		LSAPT::		;APT EXPANSION
(4)	002036	000000	LSAPT::	.WORD 0	
(5)	002040		LSDTP::		;PTR. TO DISPATCH TABLE
(4)	002040	026052	LSDTP::	.WORD LSDISPATCH	
(5)	002042		LSPRIO::		;DIAGNOSTIC RUN PRIORITY
(4)	002042	000340	LSPRIO::	.WORD 340	
(5)	002044		LSENV1::		;FLAGS DESCRIBE HOW IT WAS SETUP
(4)	002044	000000	LSENV1::	.WORD 0	
(5)	002046		LSEXP1::		;EXPANSION WORD
(4)	002046	000000	LSEXP1::	.WORD 0	
(5)	002050		LSMREV::		;SVC REV AND EDIT #
(4)	002050	003	LSMREV::	.BYTE CSREVISION	
(3)	002051	003	LSMREV::	.BYTE CSREVISION	
(5)	002052		LSEF::		;DIAG. EVENT FLAGS
(4)	002052	000000	LSEF::	.WORD 0	
(5)	002054	000000		.WORD 0	
(5)	002056		LSSPC::		
(4)	002056	000000	LSSPC::	.WORD 0	
(5)	002060		LSDEVP::		; POINTER TO DEVICE TYPE LIST
(4)	002060	002142	LSDEVP::	.WORD LSDVTYP	
(5)	002062		LSREPP::		;PTR. TO REPORT CODE
(4)	002062	011644	LSREPP::	.WORD LSRPT	
(5)	002064		LSEXP4::		
(4)	002064	000000	LSEXP4::	.WORD 0	
(5)	002066		LSEXP5::		
(4)	002066	000000	LSEXP5::	.WORD 0	
(5)	002070		LSAUT::		;PTR. TO ADD UNIT CODE
(4)	002070	007432	LSAUT::	.WORD LSAU	
(5)	002072		LSDUT::		;PTR. TO DROP UNIT CODE
(4)	002072	007350	LSDUT::	.WORD LSDU	
(5)	002074		LSLUN::		;LUN FOR EXERCISERS TO FILL
(4)	002074	000000	LSLUN::	.WORD 0	
(5)	002076		LSDESP::		;POINTER TO DIAG. DESCRIPTION
(4)	002076	002122	LSDESP::	.WORD LSDESC	
(5)	002100		LSLOAD::		;GENERATE SPECIAL AUTOLOAD EMT
(4)	002100	104035	LSLOAD::	EMT ESLOAD	
(5)	002102		LSETP::		;POINTER TO ERR_TBL
(4)	002102	000000	LSETP::	.WORD 0	
(5)	002104		LSICP::		;PTR. TO INIT CODE
(4)	002104	006374	LSICP::	.WORD LSINIT	
(5)	002106		LSCCP::		;PTR. TO CLEAN-UP CODE
(4)	002106	007272	LSCCP::	.WORD LSCLEAN	
(5)	002110		LSACP::		;PTR. TO AUTO CODE
(4)	002110	007214	LSACP::	.WORD LSAUTO	
(5)	002112		LSPRT::		;PTR. TO PROTECT TABLE
(4)	002112	006366	LSPRT::	.WORD LSPROT	
(5)	002114		LSTEST::		;TEST NUMBER
(4)	002114	000000	LSTEST::	.WORD 0	
(5)	002116		LSDLY::		;DELAY COUNT
(4)	002116	000000	LSDLY::	.WORD 0	

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 ^{F 2} PAGE 2-2
PROGRAM HEADER

SEQ 0018

(5)	002120					LSHIME::		;PTR. TO HIGH MEM
(4)	002120	000000					.WORD 0	
43								
44	002122						DESCRIPT <RLV12 DISKLESS>	
(4)	002122					LSDESC::		
(3)	002122	046122	030526	020062			.ASCIZ /RLV12 DISKLESS/	
(3)	002130	044504	045523	042514				
(3)	002136	051523	000					
(2)		002142					.EVEN	
45								
46	002142						DEV TYP <RLV12, RLV11, OR RL11/	
(4)	002142					LSDVTYP::		
(3)	002142	046122	030526	026062			.ASCIZ /RLV12, RLV11, OR RL11/	
(3)	002150	051040	053114	030461				
(3)	002156	020054	051117	051040				
(3)	002164	030514	000061					
(2)							.EVEN	
47								
48		000001				SVCGBL= 1		; SHOVE EVERYTHING BACK TO THE RIGHT.
49		000001				SVCINS= 1		

```

51          .SBTTL  HARDWARE PARAMETER CODING
52          .
53          .GET PARAMETERS FROM OPERATOR.
54          .
55          BGNHRD
56          002170 000036
57          002172
58          002172
59          002172 004130
60          002172 002266
61          002176 000002
62          002200
63          002200 005044
64          002202
65          002202 004130
66          002204 002276
67          002206 000001
68          002210
69          002210 004004
70          002212
71          002212 004130
72          002214 002313
73          002216 000001
74          002220
75          002220 000031
76          002222 002323
77          002224 160000
78          002226 177776
79          002230
80          002230 001031
81          002232 002340
82          002234 000000
83          002236 000776
84          002240
85          002240 002032
86          002242 002351
87          002244 000340
88          002246 000000
89          002250 000007
90          002252
91          002252 006004
92          002254
93          002254 003032
94          002256 002363
95          002260 003400
96          002262 000000
97          002264 000007
98          002266
99          002266
100          002266
101          002266
102          002266
103          002266
104          002266
105          002266
106          002266
107          002266
108          002266
109          002266
110          002266
111          002266
112          002266
113          002266
114          002266
115          002266
116          002266
117          002266
118          002266
119          002266
120          002266
121          002266
122          002266
123          002266
124          002266
125          002266
126          002266
127          002266
128          002266
129          002266
130          002266
131          002266
132          002266
133          002266
134          002266
135          002266
136          002266
137          002266
138          002266
139          002266
140          002266
141          002266
142          002266
143          002266
144          002266
145          002266
146          002266
147          002266
148          002266
149          002266
150          002266
151          002266
152          002266
153          002266
154          002266
155          002266
156          002266
157          002266
158          002266
159          002266
160          002266
161          002266
162          002266
163          002266
164          002266
165          002266
166          002266
167          002266
168          002266
169          002266
170          002266
171          002266
172          002266
173          002266
174          002266
175          002266
176          002266
177          002266
178          002266
179          002266
180          002266
181          002266
182          002266
183          002266
184          002266
185          002266
186          002266
187          002266
188          002266
189          002266
190          002266
191          002266
192          002266
193          002266
194          002266
195          002266
196          002266
197          002266
198          002266
199          002266
200          002266
201          002266
202          002266
203          002266
204          002266
205          002266
206          002266
207          002266
208          002266
209          002266
210          002266
211          002266
212          002266
213          002266
214          002266
215          002266
216          002266
217          002266
218          002266
219          002266
220          002266
221          002266
222          002266
223          002266
224          002266
225          002266
226          002266
227          002266
228          002266
229          002266
230          002266
231          002266
232          002266
233          002266
234          002266
235          002266
236          002266
237          002266
238          002266
239          002266
240          002266
241          002266
242          002266
243          002266
244          002266
245          002266
246          002266
247          002266
248          002266
249          002266
250          002266
251          002266
252          002266
253          002266
254          002266
255          002266
256          002266
257          002266
258          002266
259          002266
260          002266
261          002266
262          002266
263          002266
264          002266
265          002266
266          002266
267          002266
268          002266
269          002266
270          002266
271          002266
272          002266
273          002266
274          002266
275          002266
276          002266
277          002266
278          002266
279          002266
280          002266
281          002266
282          002266
283          002266
284          002266
285          002266
286          002266
287          002266
288          002266
289          002266
290          002266
291          002266
292          002266
293          002266
294          002266
295          002266
296          002266
297          002266
298          002266
299          002266
300          002266
301          002266
302          002266
303          002266
304          002266
305          002266
306          002266
307          002266
308          002266
309          002266
310          002266
311          002266
312          002266
313          002266
314          002266
315          002266
316          002266
317          002266
318          002266
319          002266
320          002266
321          002266
322          002266
323          002266
324          002266
325          002266
326          002266
327          002266
328          002266
329          002266
330          002266
331          002266
332          002266
333          002266
334          002266
335          002266
336          002266
337          002266
338          002266
339          002266
340          002266
341          002266
342          002266
343          002266
344          002266
345          002266
346          002266
347          002266
348          002266
349          002266
350          002266
351          002266
352          002266
353          002266
354          002266
355          002266
356          002266
357          002266
358          002266
359          002266
360          002266
361          002266
362          002266
363          002266
364          002266
365          002266
366          002266
367          002266
368          002266
369          002266
370          002266
371          002266
372          002266
373          002266
374          002266
375          002266
376          002266
377          002266
378          002266

```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30C(1063) 08-FEB-82 10:07 H² PAGE 3-1
HARDWARE PARAMETER CODING

SEQ 0020

73	002323	103	051123	040440	CSRA:	.ASCII	'CSR ADDRESS	:
74	002340	042526	052103	051117	VECA:	.ASCII	'VECTOR	:
75	002351	102	020122	042514	BRL:	.ASCII	'BR LEVEL	:
76	002363	104	044522	042526	DRN:	.ASCII	'DRIVE	:
77		002374				.EVEN		: UNUSED.

....AND THESE ARE THE SINGLE UNIT DEFAULTS (HARD P-TABLE).

81	002374	
(3)	002374	000005
(3)	002376	
82	002376	174400
83	002400	000160
84	002402	000200
85	002404	000000
86	002406	000003

BGNHW

.WORD	174400
.WORD	160
.WORD	PRI04
.WORD	C
.WORD	3

LSHW:: .WORD L10001-LSHW/2

:	(0)	CSR BUS ADDRESS.
:	(2)	VECTOR
:	(4)	PRIORITY
:	(6)	DRIVE (BITS 8,9,10)
:	(10)	CONTROLLER TYPE...
:	0	= RLV11...
:	1	= RLV11...
:	2	= RLV12, BAE DISABLED...
:	3	= RLV12, BAE ENABLED.

91	002410
(3)	002410

ENDHW

L10001:

```

93      .SBTTL SOFTWARE PARAMETER CODING
94      :
95      : GET SOFT PARAMETERS FROM OPERATOR.
96      :
97      BGNSFT
98      (3) 002410 000054          .WORD L10002-L$SOFT/2
99      (3) 002412          L$SOFT::
100     (4) 002412 000052          GPRMD ADLIM,0,D,-1,0,-1,YES ; GET AUTO-DROP ERROR LIMIT.
101     (4) 002414 002542          .WORD T$CODE
102     (4) 002416 177777          .WORD ADLIM
103     (4) 002420 000000          .WORD -1
104     (4) 002422 177777          .WORD T$LOLIM
105     (4) 002424          .WORD T$HILIM
106     (4) 002424 000003          DISPLAY TLMBP ; BYPASS MESSAGE.
107     (4) 002426 002575          .WORD T$CODE
108     (4) 002430          .WORD TLMBP
109     (4) 002430 005130          GPRML TLMIN,12,BIT0,YES ; TLM ON THIS UNIT ??
110     (4) 002432 002751          .WORD T$CODE
111     (4) 002434 000001          .WORD TLMIN
112     (4) 002436          .WORD BIT0
113     (5) 002436 042044          XFERF 1$
114     (4) 002440          .WORD T$CODE
115     (4) 002440 006031          GPRMA TCSU0,14,0,160000,177776,YES ; TLM CSR FOR UNIT 0.
116     (4) 002442 002776          .WORD T$CODE
117     (4) 002444 160000          .WORD TCSU0
118     (4) 002446 177776          .WORD T$LOLIM
119     (4) 002450          .WORD T$HILIM
120     (4) 002450 001130          GPRML PRMDMP,2,BIT0,YES ; PROM DUMP ??
121     (4) 002452 005024          .WORD T$CODE
122     (4) 002454 000001          .WORD PRMDMP
123     (4) 002456          .WORD BIT0
124     (5) 002456 032024          XFERT 1$
125     (4) 002460          .WORD T$CODE
126     (4) 002460 002032          GPRMD PRM,4,0,-1,0,-1,YES ; PROM ID.
127     (4) 002462 003047          .WORD T$CODE
128     (4) 002464 177777          .WORD PRM
129     (4) 002466 000000          .WORD -1
130     (4) 002470 177777          .WORD T$LOLIM
131     (4) 002472          .WORD T$HILIM
132     (4) 002472 003130          GPRML INHMP,6,BIT15,YES ; INHIBIT MAX PEAK...
133     (4) 002474 003070          .WORD T$CODE
134     (4) 002476 100000          .WORD INHMP
135     (4) 002500          .WORD BIT15
136     (4) 002500 003130          GPRML INHNP,6,BIT14,YES ; ...MIN PEAK...
137     (4) 002502 003120          .WORD T$CODE
138     (4) 002504 040000          .WORD INHNP
139     (4) 002506          .WORD BIT14
140     (4) 002506 003130          GPRML INHNC,6,BIT2,YES ; ...NOMINAL CLOCK...
141     (4) 002510 003150          .WORD T$CODE
142     (4) 002512 000004          .WORD INHNC
143     (4) 002514          .WORD BIT2
144     (4) 002514 003130          GPRML INHFC,6,BIT1,YES ; ...FAST CLOCK...
145     (4) 002516 003177          .WORD T$CODE
146     (4) 002520 000002          .WORD INHFC
147     (4) 002522          .WORD BIT1
148     (4) 002522          GPRML INHSC,6,BIT0,YES ; ...SLOW CLOCK.

```

SEQ 0022

```

.WORD      TSCODE
.WORD      INHSC
.WORD      BIT0

.WORD      TSCODE
.WORD      SEC1
.WORD      7
.WORD      T$LOLIM
.WORD      T$HILIM

```

18: ENDSFT

L10002:

```

ADLIM: .ASCII 'ERROR LIMIT FOR AUTO-DROP '
TLMBP: .ASCII 'ALL REMAINING QUERIES ARE FOR OPTIONAL (MANUFACTURING)'
       .ASCII 'G5388 TEST-LOOP-MODULE SET-UP.  USE <^Z> TO BYPASS.'
       .BYTE 0
TLMIN: .ASCII 'G5388 TLM INSTALLED '
TCSUO: .ASCII 'CSR ADDRESS (UNIT 0) '
PRMDMP: .ASCII 'DUMP PROM ON TTY: '
PRM: .ASCII 'PROM ID NUMBER '
INHXP: .ASCII 'INHIBIT MAX PEAK SHIFT '
INHMP: .ASCII 'INHIBIT MIN PEAK SHIFT '
INHNC: .ASCII 'INHIBIT NOMINAL CLOCK '
INHFC: .ASCII 'INHIBIT FAST CLOCK '
INHSC: .ASCII 'INHIBIT SLOW CLOCK '
SEC1: .ASCII 'SECTOR NUMBER (0=USE ALL)'
      .EVEN

```

... AND THESE ARE THE DEFAULT SOFT PARAMETERS.

```

      BGNSW
                                .WORD    L10003-LSSW/2
                                LSSW::
ERRLMT: .WORD    0              ; (00) NZ = DROP AFTER N ERRORS.
PDSW:   .WORD    0              ; (02) PROM DUMP SWITCH (BIT0).
PROMID: .WORD    401            ; (04) CURRENT PROM SET ID.
MPXCLK: .WORD    0              ; (06) PEAK SHIFT AND CLOCK INHIBITS.
SNGLSEC: .WORD    0             ; (10) SINGLE SECTOR TO USE (0 = ALL 6).
TLMF:   .WORD    0              ; (12) G5388 TLM FLAG (1 = INSTALLED).
:CS0:   .WORD    160010         ; (14) G5388 TLM CSR ADDRESS.
      ENDSW
                                I10003:

```

L10003:

```

145      .SBTTL  GLOBAL EQUATES
146
147      003322      EQUALS      ; GET STANDARD EQUATES.
(1)      ;
(1)      ; BIT DIFINITIONS
(1)      ;
(1)      100000      BIT15== 100000
(1)      040000      BIT14== 40000
(1)      020000      BIT13== 20000
(1)      010000      BIT12== 10000
(1)      004000      BIT11== 4000
(1)      002000      BIT10== 2000
(1)      001000      BIT09== 1000
(1)      000400      BIT08== 400
(1)      000200      BIT07== 200
(1)      000100      BIT06== 100
(1)      000040      BIT05== 40
(1)      000020      BIT04== 20
(1)      000010      BIT03== 10
(1)      000004      BIT02== 4
(1)      000002      BIT01== 2
(1)      000001      BIT00== 1
(1)      ;
(1)      001000      BIT9==  BIT09
(1)      000400      BIT8==  BIT08
(1)      000200      BIT7==  BIT07
(1)      000100      BIT6==  BIT06
(1)      000040      BIT5==  BIT05
(1)      000020      BIT4==  BIT04
(1)      000010      BIT3==  BIT03
(1)      000004      BIT2==  BIT02
(1)      000002      BIT1==  BIT01
(1)      000001      BIT0==  BIT00
(1)      ;
(1)      ; EVENT FLAG DEFINITIONS
(1)      ; EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
(1)      ;
(1)      000040      EF.START== 32.      ; START COMMAND WAS ISSUED
(1)      000037      EF.RESTART== 31.      ; RESTART COMMAND WAS ISSUED
(1)      000036      EF.CONTINUE== 30.      ; CONTINUE COMMAND WAS ISSUED
(1)      000035      EF.NEW== 29.      ; A NEW PASS HAS BEEN STARTED
(1)      000034      EF.PWR== 28.      ; A POWER-FAIL/POWER-UP OCCURRED
(1)      ;
(1)      ; PRIORITY LEVEL DEFINITIONS
(1)      ;
(1)      000340      PRI07== 340
(1)      000300      PRI06== 300
(1)      000240      PRI05== 240
(1)      000200      PRI04== 200
(1)      000140      PRI03== 140
(1)      000100      PRI02== 100
(1)      000040      PRI01== 40
(1)      000000      PRI00== 0
(1)      ;
(1)      ; OPERATOR FLAG BITS

```



```

(1)      000004      EVL==      4
(1)      000010      LOT==      10
(1)      000020      ADR==      20
(1)      000040      IDU==      40
(1)      000100      ISR==     100
(1)      000200      UAM==     200
(1)      000400      BOE==     400
(1)      001000      PNT==    1000
(1)      002000      FRI==    2000
(1)      004000      IXE==    4000
(1)      010000      IBE==   10000
(1)      020000      IER==   20000
(1)      040000      LOE==   40000
(1)      100000      HOE==  100000
148      ;
149      ; RL/RLV EQUATES.
150      ;
151      000004      ERRVEC= 4      ; BUS-ERROR VECTOR.
152      000000      RL11= 0      ; EQUATE THE CONTROLLER TYPES.
153      000001      RLV11= 1
154      000002      RLV12= 2      ; RLV12 WITH BAE DISABLED.
155      000003      RLV12X= 3      ; RLV12 WITH BAE ENABLED.
156      ;
157      100000      ERR= BIT15      ; COMPOSITE ERROR (RLCS)
158      040000      DERR= BIT14      ; DRIVE ERROR (RLCS)
159      020000      NXM= BIT13      ; NON-EXISTANT MEMORY (IF OPI = 0)...
160      020000      PAR= BIT13      ; ...OR PARITY ERROR (IF OPI = 1).
161      010000      DLT= BIT12      ; DATA LATE ERROR (IF OPI=0)...
162      010000      HNF= BIT12      ; ...OR HEADER NOT FOUND (IF OPI=1).
163      004000      DCRC= BIT11      ; DATA CRC ERROR (IF OPI=0)...
164      004000      HCRC= BIT11      ; ...OR HEADER CRC ERROR (IF OPI=1).
165      002000      OPI= BIT10      ; OPERATION INCOMPLETE (RLCS)
166      000000      DS0= 0      ; DRIVE SELECT 0 (RLCS)
167      000400      DS1= BIT8      ; DRIVE SELECT 1 (RLCS)
168      001000      DS2= BIT9      ; DRIVE SELECT 2 (RLCS)
169      001400      DS3= BIT8!BIT9 ; DRIVE SELECT 3 (RLCS)
170      000200      CRDY= BIT7      ; CONTROLLER READY (RLCS)
171      000100      INTEN= BIT6      ; INTERRUPT ENABLE (RLCS)
172      000000      NOOP= 0      ; (0) NO, P -- RL11 ONLY.
173      000000      MAINT= 0      ; (0) MAINTENANCE -- RLV11 AND RLV12.
174      000002      WRCHK= BIT1      ; (1) WRITE CHECK FUNCTION
175      000004      GSTAT= BIT2      ; (2) GET STATUS FUNCTION
176      000006      SEEK= BIT2!BIT1 ; (3) SEEK FUNCTION
177      000010      RDHDR= BIT3      ; (4) READ HEADER FUNCTION
178      000012      WRITE= BIT3!BIT1 ; (5) WRITE DATA FUNCTION
179      000014      READ= BIT3!BIT2  ; (6) READ DATA FUNCTION
180      000016      RDNHDR= BIT3!BIT2!BIT1 ; (7) READ DATA WITHOUT HEADER CHECK.
181      000001      DRDY= BIT0      ; DRIVE READY.
182      ;
183      000010      DRST= BIT3      ; DRIVE RESET (RLDA)
184      000002      GSBIT= BIT1      ; GET STATUS BIT (RLDA)
185      000001      MK= BIT0      ; MARKER BIT (RLDA)
186      000004      SIGN= BIT2      ; SIGN BIT (RLDA)
187      000100      RHHS= BIT6      ; HEAD SELECT IN READ HEADER
188      000100      STHS= BIT6      ; HEAD SELECT IN STATUS BACK

```


CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 M² PAGE 5-2
GLOBAL EQUATES

SEQ 0025

189

000020

DAHS= BIT4

;HEAD SELECT IN SEEK

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 N² PAGE 5-3
GLOBAL EQUATES

SEQ 0026

```
191  
192  
193  
194  
195  
196      000000  
197      000143  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217      000401  
218      000402  
219      000403
```

```
:  
: A COUPLE OF MACROS TO REDEFINE THE ERROR CALLING CONVENTIONS  
: SO THAT SEQUENTIAL ERRORS ARE ASSIGNED AT ASSEMBLY TIME,  
: AND I DON'T HAVE TO SCREW AROUND WITH THEM !!  
:  
SFN= 0 ; SYSTEM FATAL ERRORS RUN FROM 1 TO 99.  
DFN= 99. ; DEVICE FATAL ERRORS RUN FROM 100. ON.  
:  
.MACRO SFERR ADDR,PNTN  
SFN=SFN+1  
ERRSF SFN,ADDR,PNTN  
.ENDM  
:  
.MACRO DFERR ADDR,PNTN  
DFN=DFN+1  
ERRDF DFN,ADDR,PNTN  
.ENDM  
:  
: ANOTHER TO START EACH TEST.  
:  
.MACRO BEGIN.TEST  
BGNTST  
.ENDM  
:  
: AND A COUPLE OF HANDY SKIPS.  
:  
SKP1= BR+1  
SKP2= BR+2  
SKP3= BR+3
```

221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276

.SBTTL GLOBAL DATA

UNITST: .WORD 0 ; CURRENT UNIT UNDER TEST.
RLCS: .WORD ; REGISTER ADDRESSES.
RLBA: .WORD
RLDA: .WORD
RLMP: .WORD
RLBAE: .WORD
BCSR: .WORD ; BASE ADDRESS
BPRIOR: .WORD
BVEC: .WORD
TCSR: .WORD ; G5388 TLM CSR ADDRESS.
DRIVE: .WORD
B.CS: .WORD ; REGISTERS BEFORE FUNCTION.
B.BA: .WORD
B.DA: .WORD
B.MP: .WORD
B.BAE: .WORD
DERFLG: .WORD
E.CS: .WORD ; REGISTERS AFTER FUNCTION.
E.BA: .WORD
E.DA: .WORD
E.MP: .WORD
E.MP1: .WORD ; MAINT MODE CRC CHECK ONLY.
E.BAE: .WORD
CPUTYP: .WORD ; 0 = 11/03, +NZ = 11/23, -1 = UNKNOWN.P
RLTYP: .WORD ; RL TYPE -- 0 = RL11, 1 = RLV11, 2(3) = RLV12.
XPOLY: .WORD 120001
BCCFBK: .WORD ; LOCATION USED BY 'SIMBCC'
CALBCC: .WORD ; LOCATION USED BY 'SIMBCC'
TEMP2: .WORD ; LOCATION USED BY 'SIMBCC'
TEMP3: .WORD ; LOCATION USED BY 'SIMBCC'
TEMP4: .WORD ; LOCATION USED BY 'SIMBCC'
TEMP5: .WORD ;
TEMP1: .WORD ; DITTO
TMP0: .WORD ; DITTO
TMP1: .WORD
TMP2: .WORD
GDDAT: .WORD
BDDAT: .WORD
INIMP: .WORD ; HOLDS INITIAL MP WORD COUNT.
INIDA: .WORD ; HOLDS CURRENT DA TEST WORD.
GDCRC3: .WORD ; HOLDS CRC OF DA+3
GDCRC4: .WORD ; HOLDA CRC OF CRC OF DA+4
OPIW: .WORD 155.
OPIX: .WORD 650.
HINUM: .WORD 176543
LONUM: .WORD 123456
TENLO: .WORD
TENHI: .WORD
DLYCNT: .WORD ; 1 MSEC DELAY COUNTER.
DROPPED: .WORD ; UNITS DROPPED COUNTER.
ERPOINT: .WORD ; POINTS TO ONE OF THE FOLLOWING:
ERCOUNT: .BLKW 64. ; ERROR COUNTERS (ENOUGH FOR 64 UNITS).
; PATTERNS USED FOR LOADING/READING REGISTERS

277				
278	003670	000000	BEGPAT: 000000	:GROWING 1
279	003672	000000	000000	
280	003674	000000	000000	
281	003676	000000	000000	
282	003678	000000	000000	
283	003680	000000	000000	
284	003682	000000	000000	
285	003684	000000	000000	
286	003686	000000	000000	
287	003688	000000	000000	
288	003690	000000	000000	
289	003692	000000	000000	
290	003694	000000	000000	
291	003696	000000	000000	
292	003698	000000	000000	
293	003700	000000	000000	
294	003702	000000	000000	
295	003704	000000	000000	
296	003706	000000	000000	
297	003708	000000	000000	
298	003710	000000	000000	
299	003712	000000	000000	
300	003714	000000	000000	
301	003716	000000	000000	
302	003718	000000	000000	
303	003720	000000	000000	
304	003722	000000	000000	
305	003724	000000	000000	
306	003726	000000	000000	
307	003728	000000	000000	
308	003730	000000	000000	
309	003732	000000	000000	
310	003734	000000	000000	
311	003736	000000	000000	
312	003738	000000	000000	
313	003740	000000	000000	
314	003742	000000	000000	
315	003744	000000	000000	
316	003746	000000	000000	
317	003748	000000	000000	
318	003750	000000	000000	
319	003752	000000	000000	
320	003754	000000	000000	
321	003756	000000	000000	
322	003758	000000	000000	
323	003760	000000	000000	
324	003762	000000	000000	
325	003764	000000	000000	
326	003766	000000	000000	
327	003768	000000	000000	
328	003770	000000	000000	
329	003772	000000	000000	
330	003774	000000	000000	
331	003776	000000	000000	
332	004000	000000	000000	
333	004002	000000	000000	
334	004004	000000	000000	
335	004006	000000	000000	
336	004008	000000	000000	
337	004010	000000	000000	
338	004012	000000	000000	
339	004014	000000	000000	
340	004016	000000	000000	
341	004018	000000	000000	
342	004020	000000	000000	
343	004022	000000	000000	
344	004024	000000	000000	
345	004026	000000	000000	
346	004028	000000	000000	
347	004030	000000	000000	
348	004032	000000	000000	
349	004034	000000	000000	
350	004036	000000	000000	
351	004038	000000	000000	
352	004040	000000	000000	
353	004042	000000	000000	
354	004044	000000	000000	

333	004046	177737	177737
334	004050	177677	177677
335	004052	177577	177577
336	004054	177377	177377
337	004056	176777	176777
338	004060	175777	175777
339	004062	173777	173777
340	004064	167777	167777
341	004066	157777	157777
342	004070	137777	137777
343	004072	077777	077777
344	004074	177777	177777
345	004076	000000	000000

ENDPAT: 000000

... PATTERNS FOR TESTING THE READ/WRITE BITS <9:1> OF THE CSR.

348				
349	004100	000000	CSPAT: .WORD 0	:SHIFTING 1
350	004102	000002	.WORD BIT1	
351	004104	000004	.WORD BIT2	
352	004106	000010	.WORD BIT3	
353	004110	000020	.WORD BIT4	
354	004112	000040	.WORD BIT5	
355	004114	000100	.WORD BIT6	
356	004116	000400	.WORD BIT8	
357	004120	001000	.WORD BIT9	
358	004122	001576	.WORD 1576	:GROWING 0
359	004124	001574	.WORD 1574	
360	004126	001570	.WORD 1570	
361	004130	001560	.WORD 1560	
362	004132	001540	.WORD 1540	
363	004134	001500	.WORD 1500	
364	004136	001400	.WORD 1400	
365	004140	001576	.WORD 1576	:SHIFT 0
366	004142	001574	.WORD 1574	
367	004144	001566	.WORD 1566	
368	004146	001556	.WORD 1556	
369	004150	001536	.WORD 1536	
370	004152	001436	.WORD 1436	
371	004154	001136	.WORD 1136	
372	004156	000076	.WORD 76	
373	004160	000006	.WORD 6	:GROWING 1
374	004162	000016	.WORD 16	
375	004164	000036	.WORD 36	
376	004166	000076	.WORD 76	
377	004170	000176	.WORD 176	:
378	004172	000576	.WORD 576	
379	004174	001576	.WORD 1576	
380	004176	000000	.WORD 0	

CSEND: .WORD 0

... TABLE OF DA TEST WORDS FOR MAINTENANCE MODE CRC CALCULATIONS.

381			PATCRC: 155552
382			155550
383			155530
384	004200	155552	066663
385	004202	155530	125247
386	004204	066663	125247
387	004206	125247	052522
388	004210	052522	

389	004212	177774	177774
390	004214	000374	000374
391	004216	022217	022217
392	004220	044441	044441
393	004222	166663	166663
394	004224	144441	144441
395	004226	033330	033330
396	004230	0111106	0111106
397	004232	070704	070704
398	004234	107065	107065
399	004236	1111106	1111106
400	004240	167353	167353
401	004242	156732	156732
402	004244	146311	146311
403	004246	135670	135670
404	004250	114626	114626
405	004252	104205	104205
406	004254	073564	073564
407	004256	063143	063143
408	004260	042101	042101
409	004262	031460	031460
410	004264	021037	021037
411	004266	010416	010416
412	004270	000000	000000

CRCEND: 000000 ; 28 TEST WORDS IN THE TABLE.

... TABLE OF DATA PATTERNS FOR MAINTENANCE MODE FIFO TESTS.

413			
414			
415			
416	004272	155555	155555
417	004274	133333	133333
418	004276	066666	066666
419	004280	125252	125252
420	004282	052525	052525
421	004284	177777	177777
422	004286	000000	000000
423	004290	107070	107070
424	004292	070707	070707
425	004294	144444	144444
426	004296	033333	033333
427	004300	011111	011111
428	004302	022222	022222
429	004304	044444	044444
430	004306	111111	111111
431	004310	166666	166666
432	004312	010421	010421
433	004314	021042	021042
434	004316	031463	031463
435	004320	042104	042104
436	004322	063146	063146
437	004324	073567	073567
438	004326	104210	104210
439	004330	114631	114631
440	004332	135673	135673
441	004334	146314	146314
442	004336	156735	156735
443	004360	167356	167356
444	004362	000000	000000

ENDDAT: 000000 ; 28 OF THESE TOO.

CVRLBA0 -- RLV12 DISKLESS MACY11 30G(1063) C3-FEB-82 10:07 F 3
CVRLBA.P11 08-FEB-82 09:59 GLOBAL DATA PAGE 6-4

SEQ 0031

445
446
447
448
449
450

004364 000400
005364 000400
006364 000000

:
: BUFFERS FOR RLV11 MAINTENANCE FUNCTION

BUF1: .BLKW 256.
BUF2: .BLKW 256.
BUFEND: 0

: MAINTENANCE MODE, DMA BUF1 => FIFO...
: ...AND FIFO => BUF2.

```

452          .SBTTL  INITIALIZATION CODE
453          :
454          : LOAD DEVICE PROTECTION NOT REQUIRED.
455          :
456          BGNPROT
457          :
458          : CSR OFFSET MAKE NOP
459          : MASS BUS OFFSET MAKE NOP
460          : DRIVE OFFSET MAKE NOP
461          :
462          : ENTER HERE FROM DRS ON START, RESTART, CONTINUE, ETC...
463          :
464          BGNINIT
465          :
466          : POWER UP ??
467          :
468          : RESTART ??
469          :
470          : NEW PASS ??
471          :
472          : CONTINUE ??
473          :
474          : NONE OF THE ABOVE.
475          :
476          :
477          :
478          :
479          :
480          :
481          :
482          :
483          :
484          :
485          :
486          :
487          :
488          :
489          :
490          :

```

006366 177777
006370 177777
006372 177777
006374

006374
(3) 006374
006374 012700 000034
(3) 006400 104447
006402
(2) 006402 103423
006404
(3) 006404 012700 000040
(3) 006410 104447
006412
(2) 006412 103417
006414
(3) 006414 012700 000037
(3) 006420 104447
006422
(2) 006422 103413
006424
(3) 006424 012700 000035
(3) 006430 104447
006432
(2) 006432 103472
006434
(3) 006434 012700 000036
(3) 006440 104447
006442
(2) 006442 103401
006444 000506
006446 000137 007050

006452
006452 000005
006454 005737 003306
006460 001402
006462 000137 024606
006464 012737 176543 003452
006474 012737 123456 003454
006502 005037 003464
006504 012701 003470
006512 012700 000100
006516 005021
006520 005300
006522 001375

1\$: BR NXTU
JMP CCNT

START:RESTART:
RESET
TST PDSW
BEQ 1\$
JMP ROMDUMP
MOV #176543,H1NUM
MOV #123456,LONUM
CLR DROPPED
MOV #ERCOUNT,R1
MOV #64,R0
CLR (R1)+
DEC R0
BNE 2\$

2\$: CLR (R1)+
DEC R0
BNE 2\$

1\$: PROM DUMP ??
BR IF NOT.
*** PRGM DUMPER, NO RETURN ***
RANDOM GEN. PRIMES.
CLEAR UNITS DROPPED COUNTER.
GET ERROR COUNTER POINTER.
CLEAR ALL ERROR COUNTERS.

491	006524	013746	000004		ENVIRN:	MOV	#4, -(SP)	:	SAVE VECTOR 4...		
492	006524	013746	000010			MOV	#10, -(SP)	:	...AND 10...		
493	006524	010605				MOV	SP, R5	:	...AND THE STACK POINTER.		
494	006524	012737	006574	000004		MOV	#LSI, #4	:	SET BUS TRAP.		
495	006524	005000				CLR	R0	:			
496	006524	005737	177776			TST	#177776	:	PSW ADDRESS VALID ??		
497	006524	000240				NOP		:	TRAP THRU 4 IF NOT (LSI11 OR 11/2).		
498	006524	012737	006570	000010		MOV	#NLSI, #10	:	SET ILLEGAL TRAP.		
499	006524	000007				MFPT=	7	:			
500	006562	000007				MFPT		:	GET PROCESSOR TYPE (IF POSSIBLE).		
501	006562	000240				NOP		:	TRAP THRU 10 IF NOT (NOT 11/23).		
502	006562	000402				SKP2		:			
503	006570	012700	177777		NLSI:	MOV	#-1, R0	:	CPU TYPE UNKNOWN.		
504	006574	010037	003400		LSI:	MOV	R0, (PUTYP	:	CPU IS LSI SOMETHING 'R OTHER.		
505	006600	010506				MOV	R5, SP	:	RESTORE STACK...		
506	006602	012637	000010			MOV	(SP)+, #10	:	...AND VECTORS.		
507	006606	012637	000004			MOV	(SP)+, #4	:			
508	006612	004737	011242			JSR	PC, .SIZE	:	SIZE AVAILABLE MEMORY.		
509	006616					DORPT		:	REPORT OPERATING ENVIRONMENT.		
510	(3)	006616	104424					:	TRAP	CSDRPT	
511	006620	023737	003464	002012	NEWPAS:	CMP	DROPPED, LSUNIT	:	UNITS STILL ALIVE ??		
512	006626	002401				BLT	1\$	:			
513	006630					DOCLN		:	NO, ABORT		
514	006630	104444						:	TRAP	CSDCLN	
515	006632	012737	003466	003466	1\$:	MOV	#ERCOUNT-2, ERPOINT	:	INIT THE UNIT ERROR POINTER.		
516	006640	013737	003454	003456		MOV	LONUM, TEMLO	:			
517	006646	013737	003452	003460		MOV	HINUM, TEMHI	:	NEW PRIMES FOR NEW PASS.		
518	006654	012737	177777	003322		MOV	#-1, UNITST	:	RESET UNIT NUMBER.		
519	006662	005237	003322		NXTU:	INC	UNITST	:	BUMP UNIT NUMBER...		
520	006666	062737	000002	003466		ADD	#2, ERPOINT	:	...AND ERROR COUNT POINTER.		
521	006674	023737	003322	002012		CMP	UNITST, LSUNIT	:			
522	006702	002346				BGE	NEWPAS	:			
523	006704				1\$:	GPHARD	UNITST, R0	:			
524	(3)	006704	013700	003322				:	MOV	UNITST, R0	
525	(3)	006710	104442					:	TRAP	C\$GPHRD	
526	006712					BNCOMPLETE	NXTU	:	BR IF UNIT NOT THERE (DROPPED).		
527	006712	103363						:	BCC	NXTU	
528	006714	012037	003336		2\$:	MOV	(R0)+, BCSR	:	SET UP RUN TIME P-TABLE...		
529	006720	005737	036704			TST	MIMIC	:	***** IF MIMIC		
530	006724	001403				BEQ	3\$	:	*****		
531	006726	012737	037224	003336		MOV	#DUMMY, BCSR	:	***** USE DUMMY REGISTERS		
532	006734	012037	003342		3\$:	MOV	(R0)+, BVEC	:	...FOR THIS UNIT.		
533	006740	012037	003340			MOV	(R0)+, BPRIOR	:			
534	006744	012037	003346			MOV	(R0)+, DRIVE	:			
535	006750	012037	003402			MOV	(R0)+, RLTP	:	SET CONTROLLER TYPE.		
536	006754	013737	003316	003344		MOV	TLMF, TCSR	:	RUNNING WITH TLM'S ??		
537	006762	001402				BEQ	5\$	:	BR IF NOT.		
538	006764	013737	003320	003344		MOV	TCSO, TCSR	:	YES, GET BASE TLM CSR...		
539	006772	013700	003322			MOV	UNITST, R0	:	...AND CURRENT UNIT NUMBER.		
540	006776	001404				BEQ	5\$	:	BR IF UNIT IS ZERO.		
541	007000	062737	000010	003344	4\$:	ADD	#10, TCSR	:	OTHERWISE, ADJUST FOR CURRENT UNIT.		
542	007006	077004				SUB	R0, 4\$	:			
543								:			
544	007010	005737	003400		5\$:	TST	CPUTYP	:			

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 ¹3
INITIALIZATION CODE PAGE 7-2

SEQ 0034

```
542 007014 003004          BGT      6$          ; BR IF 11/23.
543 007016 001406          BEQ      7$          ; BR IF LSI 11 OR 11/2.
544 007020 012700 000764    MOV      #500.,R0    ; IF NEITHER, DELAY IS UNCALIBRATED.
545 007024 000405          BR        8$          ;
546 007026 012700 000443    6$: MOV      #291.,R0    ; 1MS DELAY FOR 11/23.
547 007032 000402          SKP2
548 007034 012700 000170    7$: MOV      #120.,R0    ; 1MS DELAY FOR LSI11 OR 11/2.
549 007040 010037 003462    8$: MOV      R0,DLYCNT    ; SET DELAY COUNTER.
550 007044 004737 026000    JSR      PC,ADJTN    ; ADJUST THE NUMBER OF TESTS TO RUN...
551                                     ; ...ACCORDING TO CURRENT CONFIGURATION.
552 007050 013737 003456 003454 CONT: MOV      TEMLO,LONUM    ; RESTORE RANDOM FOR NEXT UUT
553 007056 013737 003460 003452 MOV      TEMHI,HINUM    ; RESTORE PRIME FOR NEXT UUT
554 007064 012737 004364 010352 MOV      #BUF1,BA16    ; INIT 16 BIT BUFFER ADDRESS.
555 007072 005037 010354    CLR      BA22        ; ...AND 6 BIT EXTENSION (22 Bits).
556 007076 013700 003336    MOV      BCSR,R0
557 007102 010037 003324    MOV      R0,RLCS
558 007106 062700 000002    ADD      #2,R0
559 007112 010037 003326    MOV      R0,RLBA
560 007116 062700 000002    ADD      #2,R0
561 007122 010037 003330    MOV      R0,RLDA
562 007126 062700 000002    ADD      #2,R0
563 007132 010037 003332    MOV      R0,RLMP
564 007136 062700 000002    ADD      #2,R0
565 007142 010037 003334    MOV      R0,RLBAE
566 007146 013700 003342    MOV      BVEC,R0
567 007152 012720 010632    MOV      #RLINT,(R0)+ ; SET RL VECTOR.
568 007156 013710 003340    MOV      BPRIOR,(R0)
569 007162 012737 010624 000004 MOV      #TRAP4,ERRVEC ; SET TIME-OUT TRAP CATCHER.
570 007170 012737 000340 000006 MOV      #PRI07,ERRVEC+2
571 007176 012737 010636 000100 MOV      #CLKRT1,#100 ; NULL THE CLOCK VECTOR.
572 007204 012737 000340 000102 MOV      #PRI07,#102
573 007212          ENDINIT
(3) 007212
(3) 007212 104411          L10005: TRAP      CSINIT
574
575          ; AUTO-DROP IF FLA:ADR AND UNIT DOESN'T RESPOND.
576          ;
577          BGNAUTO
(3) 007214
578 007214 023737 003322 002012 CMP      UNITST,LSUNIT ; VALID UNIT NUMBER ??
579 007222 002022          BGE      1$          ; NO, DON'T TRY IT !!!
580 007224 005037 010626          CLR      TRPFLG    ; CLEAR TRAP FLAG.
581 007230 005777 174070          TST      @RLCS      ; TRY TO ACCESS CONTROLLER.
582 007234 000240          240
583 007236 005737 010626          TST      TRPFLG    ; DID IT TRAP ??
584 007242 001412          BEQ      1$          ; NO, WE'RE OK.
585 007244 013737 003324 003432 MOV      RLCS,GDDAT
586 007252          SFERR      EMO,ERR1    ; REPORT NO CONTROLLER...
(5) 007252 104454
(6) 007254 000001          TRAP      CSERSF
(6) 007256 030636          .WORD      1
(6) 007260 026170          .WORD      EMO
587 007262          .WORD      ERR1
(3) 007262 013700 003322          DODU      UNITST    ; ...AND TELL SUPER TO DROP UNIT.
(3) 007266 104451          MOV      UNITST,R0
588 007270          1$: ENDAUTO          TRAP      CS$DODU
```

```

(3) 007270
(3) 007270 104461
589
590
591
592 007272
(3) 007272
593 007272 012700 000012
594 007272 032777 000200 174020 1$: MOV #10.,R0
595 007304 001004 BNE 2$ ; GIVE HIM ABOUT 100 MSEC...
596 007306 004537 010160 JSR R5,WDELAY ;...TO FINISH UP.
597 007312 000012 10.
598 007314 077010 SOB R0,1$
599 007316 012777 000200 174000 2$: MOV #CRDY,ARLCS ; THEN CLEAR ALL OPTION BITS.
600 007324 023727 003402 000003 CMP RLTP,#RLV12X ; BAE IN USE ??
601 007332 001002 BNE 3$
602 007334 005077 173774 CLR ARLBAE ; IF SO, MAKE SURE IT'S CLEAR.
603 007340 3$: SETPRI #PRI07
(3) 007340 012700 000340
(3) 007344 104441
604 007346
(3) 007346
(3) 007346 104412
605
606
607
608 007350
(3) 007350
609 007350 023737 002012 003464
610 007356 001424
611 007360 005237 003464
612 007364
(7) 007364 012746 007406
(6) 007370 012746 000001
(3) 007374 010600
(4) 007376 104417
(4) 007400 062706 000004
613 007404 000411
614 007406 040445 047125 052111 1$: BR 2$
615 007430 007430 .ASCIZ /%AUNIT DROPPED%/
616 007430 2$: .EVEN
(3) 007430
(3) 007430 104453
617
618
619
620 007432
(3) 007432
621 007432 005737 003464
622 007436 001423
623 007440 005337 003464
624 007444
(7) 007444 012746 007466
(6) 007450 012746 000001
(3) 007454 010600
(4) 007456 104417

; CLEAN-UP CODING SECTION (DO AT END-PASS (OR SUB PASS)).
BGNCLN
L$CLEAN::
MOV #PRI07,R0
TRAP C$SPRI
ENDCLN
L10006: TRAP C$AUTO

; DROP UNITS -- AND KEEP TRACK OF 'EM.
BGNDU
L$DU::
CMP L$UNIT,DROPPED ; ALL UNITS DROPPED ??
BEQ 2$ ; YUP, DO NOTHING.
INC DROPPED ; NOT YET, BUMP COUNT...
PRINTF #1$ ;...AND TELL THE MAN.
MOV #1$,-(SP)
MOV #1$,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #4,SP

L10007: TRAP C$CLEAN

L10010: TRAP C$DU

; IF THEY'RE PICKED UP AGAIN, TWEAK THE COUNTER.
BGNAU
L$AU::
TST DROPPED ; ALL UNITS BACK IN PLACE ??
BEQ 2$ ; JUST EXIT IF SO.
DEC DROPPED ; NOT YET, ADJUST COUNT...
PRINTF #1$ ;...AND TELL HIM.
MOV #1$,-(SP)
MOV #1$,-(SP)
MOV SP,R0
TRAP C$PNTF

```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59 MACY11 30G(1063) 08-FEB-82 10:07 ^{K 3} PAGE 7-4
INITIALIZATION CODE

SEQ 0036

(4)	007460	062706	000004						
625	007464	000410							
626	007466	040445	047125	052111	1\$:	BR	2\$		
627		007506				.ASCIZ	/XAUNIT ADDEDXN/		
628	007506				2\$:	.EVEN			
(3)	007506					ENDAU			
(3)	007506	104452							

ADD #4,SP

L10011: TRAP CSAU

```

630      .SBTTL  GLOBAL SUBROUTINES
631
632      :SUBROUTINE TO CHECK FOR CONTROLLER ERRORS.
633      :TRANSLATE THE ERROR BITS INTO PLAIN LANGUAGE AND MERGE THEM
634      :INTO THE ERROR BUFFER (EM99) FOR SUBSEQUENT PRINTING.
635      :DRIVE ERROR IS INCLUDED, BUT IS IGNORED IN THE DISKLESS TESTS.
636
637      :CALL:  JSR      R5,GETERR
638              BR      XX      : NO ERRORS FOUND, RETURN.
639              ERRXX      : ERROR, RETURN.
640
641      GETERR: MOV      R1,-(SP)      : SAVE R1.
642              CLR      DERFLG      : CLEAR OUT DRIVE ERROR FLAG
643              MOV      #EM99,R1     : SET TEXT BUFFER POINTER.
644              BIT      #176000,E.CS : ANY ERRORS THERE ??
645              BNE      1$           : BR IF SO.
646              JSR      R5,FIX       : NO, JUST SAY 'NONE'...
647              NON
648              BR      10$           :...AND RETURN.
649
650      1$:  TST      E.CS           : IS COMPOSITE ERROR SET ? (BETTER BE)
651              BPL      2$           : IF NOT SOMETHING'S WRONG !!!
652              JSR      R5,FIX       : YES, PUT 'CERR' IN STRING.
653              CERR
654      2$:  BIT      #DERR,E..S     : DRIVE ERROR SET?
655              BEQ      3$           : BR IF NOT.
656              INC      DERFLG      : YES, SET DRV ERROR FLAG
657              JSR      R5,FIX       : ADD 'DRV' TO STRING.
658              DEMES
659      3$:  BIT      #036000 E.CS    : ANY OTHERS ???
660              BEQ      10$          : NO, IGNORE THE DRIVE ERROR (DISKLESS).
661              TST      (R5)+        : YES, BUMP PC TO TAKE ERROR RETURN.
662              BIT      #NXM!PAR,E.CS : NON-EX OR PARITY ??
663              BEQ      5$           : BR IF NOT.
664              MOV      #NXMES,4$
665              BIT      #OPI,E.CS
666              BEQ      +10
667              MOV      #PAR!ES,4$
668              JSR      R5,FIX       : ADD 'NXM' OR 'PAR'.
669              NXMES
670      4$:  BIT      #DLT!P!F,E.CS   : DATA LATE OR HEADER NOT FOUND ??
671              BEQ      7$           : BR IF NOT.
672              MOV      #DLT!S,6$
673              BIT      #OPI,E.CS
674              BEQ      +10
675              MOV      #HNF!ES,6$
676              JSR      R5,FIX       : ADD 'DLT' OR 'HNF'.
677              DLTMES
678      5$:  BIT      #DCRC!HCRC,E.CS : EITHER CRC ??
679              BEQ      9$           : BR IF NOT.
680              MOV      #DCRCES,8$
681              BIT      #OPI,E.CS
682              BEQ      +10
683              MOV      #HCRCMES,8$
684              JSR      R5,FIX       : ADD 'DCRC' OR 'HCRC'.
685      6$:  DCRMES

```

```

686 007760 032737 002000 003364 9$: BIT #OPI,E.CS ; 'OPI' SET ??
687 007766 001403 10$: BEQ 10$
688 007770 004537 010004 JSR R5,FIX ; ADD 'OPI' TO STRING.
689 007774 030623 OPIMES
690 007776 105011 10$: CLRB (R1) ; TERMINATE ERROR LIST...
691 010000 012601 MOV (SP)+,R1 ; ...AND RESTORE R1.
692 010002 000205 RTS R5 ; RETURN.
693
694 ; ROUTINE TO MOVE ASCII STRINGS
695 ; USES REGISTERS R1 - WHERE STRING IS BEING BUILT
696
697 ; CALL JSR R5,FIX
698 ; .WORD ; ADDRESS OF STRING TO MOVE
699
700 010004 012500 FIX: MOV (R5)+,R0 ; GET ADDRESS AND MOVE RETURN
701 010006 112021 1$: MOVB (R0)+,(R1)+ ; GET BYTE AND UPDATE
702 010010 001776 BNE 1$ ; WATCH 0 BYTE TERMINATOR
703 010012 105741 TSTB -(R1) ; BACK UP OVER ZERO BYTE
704 010014 000275 RTS R5 ; EXIT
705
706 ; LOAD REGISTERS BEFORE FUNCTION
707 ; CALL: JSR R5,BEFORE
708
709 010016 017737 173302 003350 BEFORE: MOV @RLCS,B.CS ; READ CS...
710 010024 042737 176000 003350 BIC #176000,B.CS ; ...ERRORS WILL BE CLEARED ON XCT.
711 010032 017737 173270 003352 MOV @RLBA,B.BA ; READ BA
712 010040 017737 173264 003354 MOV @RLDA,B.DA ; READ DA
713 010046 013737 003436 003356 MOV @RLMP,B.MP ; INITIAL MP IS THE WORD COUNT.
714 010054 023737 003402 000003 CMP RLTP,#RLV12X
715 010062 001003 BNE 1$
716 010064 017737 173274 003360 1$: MOV @RLBAE,B.EAE ; READ BAE IF ENABLED.
717 010072 000205 RTS R5
718
719 ; LOAD REGISTERS AT ERROR OR DONE.
720 ; CALL: JSR R5,AFTER
721
722 010074 017737 173224 003364 AFTER: MOV @RLCS,E.CS ; READ CS
723 010102 017737 173220 003366 MOV @RLBA,E.BA ; READ BA
724 010110 017737 173214 003370 MOV @RLDA,E.DA ; READ DA
725 010116 017737 173210 003372 MOV @RLMP,E.MP ; READ MP
726 010124 005737 036704 TST MIMIC ; ***** IF MIMIC
727 010130 001003 BNE 1$ ; ***** BYPASS 2ND MP READ
728 010132 017737 173174 003374 1$: MOV @RLMP,E.MP1 ; READ MP (MAINT 2ND CRC CHECK ONLY).
729 010140 023727 003402 000003 CMP RLTP,#RLV12X
730 010146 001003 BNE 2$
731 010150 017737 173160 003376 2$: MOV @RLBAE,E.BAE ; READ BAE IF ENABLED.
732 010156 000205 RTS R5
733
734 ; ROUTINE TO DELAY IN 1 MSEC INCREMENTS.
735 ; CALIBRATED FOR LSI'S ONLY.
736 ; DELAY COUNT IS 291. (443B) FOR LSI 11/23.
737 ; 120. (170B) FOR LSI 11 OR 11/2.
738 ; 500. (764B) IF CPU TYPE IS UNKNOWN.
739
740 ; CALL: JSR R5,WDELAY
741 ; N ; DELAY N MSECS.

```


742
743 010160 110146
744 010162 110246
745 010164 012500
746 010166 013700 003462
747 010172 005300
748 010174 001370
749 010176 005300
750 010200 001370
751 010202 012100
752 010204 012100
753 010206 000000

```
WDELAY: MOV R1, -(SP)
        MOV R2, -(SP)
        MOV (R5)+, R2
        DLYCNT, R1
2$:     DEC R1
        BNE 3$
        DEC R2
        BNE 2$
4$:     MOV (SP)+, R2
        MOV (SP)+, R1
        RTS R5
```

; APPROX MSEC DELAY
; GET 1 MSEC TIMER.
; START LOOP
; CHECK ON MSECs REQUESTED
; BRANCH AND DO ANOTHER LOOP
; SETUP FOR RETURN AFTER DELAY

; ROUTINE TO LOAD RLCS WITH RLV11/12 MAINTENANCE FUNCTION
; EITHER FLAG DRIVEN OR INTERRUPT MODE.

```
CALL:   JSR R5, LDFUN
        .WORD MAINT ; OR MAINT!INTEN
        .WORD -N ; WORD COUNT.
```

; LOCATIONS 'BA16' AND 'BA22' COMPRISE A 22 BIT PHYSICAL BUS ADDRESS
; AND MUST BE SET BY THE CALLING ROUTINE PRIOR TO ENTRY.

765 010210 012500
766 010212 012537 003436
767 010216 042700 177601
768 010222 053700 003346
769 010226 052700 000200
770 010232 010077 730666
771 010236 013777 110352
772 010244 023727 1103402 173062 010003
773 010252 001413
774 010254 013700 110354
775 010260 042700 177774
776 010264 006300
777 010266 006300
778 010270 006300
779 010272 006300
780 010274 050077 173024
781 010280 000403
782 010282 013777 010354 173024
783 010284 013777 003440 173012
784 010286 013777 003436 173006
785 010288 004537 010016
786 010290 042777 000200 172766
787 010296 005737 033704
788 010302 001402
789 010304 004737 03-706
790 010330 002205
791
792 010352 004364
793 010354 000000
794
795
796
797

```
LDFUN:  MOV (R5)+, R0
        MOV (R5)+, INIMP
        BIC #177661, R0
        BIS DRIVE, R0
        BIS #CRDY, R0
        MOV R0, @RLCS
        MOV BA16, @RLBA
        CMP RLTP, #RLV12X
        BEQ 1$
        MOV BA22, R0
        BIC #AC3, R0
        ASL R0
        ASL R0
        ASL R0
        ASL R0
        BIS R0, @RLCS
        BR 2$
1$:     MOV BA22, @RLBAE
2$:     MOV INIDA, @RLDA
        MOV INIMP, @RLMP
        JSR R5, BEFORE
        BIC #CRDY, @RLCS
        TST MIMIC
        BEQ 3$
        JSR PC, EMURLV
3$:     RTS R5
```

; BUILD THE FUNCTION IN R0.
; SAVE WORD COUNT FOR LATER.
; MASK GARBAGE BITS.
; INSERT DRIVE (0).
; ...AND THE READY BITS.
; LOAD UP THE CSR.
; LOAD BUS ADDRESS <15:0>.
; RLV12 WITH BAE ??
; BR IF SO.
; NO, GET EXTENSION IF ANY...
; ...KEEP <17:16> ONLY...
; ...MOVE 'EM TO <5:4>...
; ...AND INSERT INTO CS.
; RLV12X, SET BA EXTENSION '21:16'.
; LOAD DISK ADDRESS.
; LOAD WORD COUNT.
; SAVE STATE BEFORE FUNCTION.
; XCT MAINT FUNCTION.
; ***** IF MIMIC
; *****
; ***** EMULATE THE RLV FUNCTION
; RETURN

```
BA16:   BUF1 ; 16 BIT BUFFER ADDRESS <15:0>
BA22:   0 ; 6 BIT EXTENSION <21:16> IN <5:0>.
```

; ROUTINE TO FILL BUFFER WITH DATA PATTERNS FOR RLV MAINTENANCE.
; BUF1 IS FILLED WITH 256 WORD PATTERN (OR COMPLEMENTING PATTERN).
; BUF2 IS FILLED WITH 255 ZEROS AND 123456.

```

798
799
800
801
802 010356 005000
803 010360 000402
804 010362 012700 177777
805 010366 010146
806 010370 010246
807 010372 012537 003424
808 010376 012701 004364
809 010402 012702 000400
810 010406 013721 003424
811 010412 005700
812 010416 001402
813 010416 005137 003424
814 010422 005302
815 010424 001370
816 010426 012702 000377
817 010432 005021
818 010434 005302
819 010436 001375
820 010440 012721 123456
821 010444 012602
822 010446 012601
823 010450 000205
824
825
826
827
828
829
830 010452 010146
831 010454 010246
832 010456 012701 004364
833 010462 012702 000400
834 010466 004537 010526
835 010472 013721 003452
836 010476 005302
837 010500 001375
838 010502 012702 000377
839 010506 005021
840 010510 005302
841 010512 001375
842 010514 012721 123456
843 010520 012602
844 010522 012601
845 010524 000205
846
847
848
849
850
851
852
853

```

```

:CALL: JSR R5,SETPAT(SETCMP)
:      .WORD N ;PATTERN FOR BUFFER
SETPAT: CLR R0 ; SET NO COMPLIMENT...
        BR +6 ;...AND SKIP NEXT.
SETCMP: MOV #-1,R0 ; SET COMPLIMENT.
        MOV R1,-(SP)
        MOV R2,-(SP)
        MOV (R5)+,TMP0 ; DATA TO STORE.
        MOV #BUF1,R1 ; BUFFER POINTER.
        MOV #256,R2 ; WORD COUNT
1$: MOV TMP0,(R1)+
        TST R0 ; ARE WE COMPLIMENTING ??
        BEQ 2$ ; NO, SKIP NEXT.
        COM TMP0 ; YES, COMPLIMENT IT.
2$: DEC R2
        BNE 1$
3$: MOV #255,R2 ; NOW CLEAR THE REST OF...
        CLR (R1)+ ;...THE BUFFER.
        DEC R2
        BNE 3$
        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
SETRAN: MOV R1,-(SP)
        MOV R2,-(SP)
        MOV #BUF1,R1 ; FIRST BUFFER START
        MOV #256,R2 ; BUFFER COUNT
1$: JSR R5,RAND ; GET RANDOM NUMBER
        MOV HINUM,(R1)+ ; STORE IN BUFFER
        DEC R2
        BNE 1$
2$: MOV #255,R2 ; NOW CLEAR THE REST.
        CLR (R1)+
        DEC R2
        BNE 2$
        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.
:CALL: JSR R5,RAND ;CALL THE ROUTINE
        RETURN ;RETURN HERE THE RANDOM NUMBER
:      ;WILL BE IN HINUM,L'NUM
:

```


854 010526 010146
855 010530 010246
856 010532 010346
857 010534 010446
858 010536 010546
859 010538 010646
860 010540 010746
861 010542 010846
862 010544 010946
863 010546 011046
864 010548 011146
865 010550 011246
866 010552 011346
867 010554 011446
868 010556 011546
869 010558 011646
870 010600 011746
871 010602 011846
872 010604 011946
873 010606 012046
874 010608 012146
875 010610 012246
876 010612 012346
877 010614 012446
878 010616 012546
879 010618 012646
880 010620 012746
881 010622 012846
882 010624 012946
883 010626 013046
884 010628 013146
885 010630 013246
886 010632 013346
887 010634 013446
888 010636 013546
889 010638 013646
890 010640 013746
891 010642 013846
892 010644 013946
893 010646 014046
894 010648 014146
895 010650 014246
896 010652 014346
897 010654 014446
898 010656 014546
899 010658 014646
900 010700 014746
901 010702 014846
902 010704 014946
903 010706 015046
904 010708 015146
905 010710 015246

003454
003452
177771

003454

003452
001057

047401
003454
003452

172452

010074

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

ROUTINES TO SET A FLAG ON BUS-ERROR AND/OR RL INTERRUPT.

```
TRAP4:  INC (PC)+
TRPFLG: 0
        RTI
```

```
RLINT:  INC (PC)+
INTFLG: 0
CLKRTI: RTI
```

; USE TO DISMISS FREE-RUNNING CLOCK.

ROUTINE TO WAIT FOR CONTROLLER READY
THE 800 MSEC INTERVAL APPLIES TO LSI'S ONLY !!!
IN ANY CASE, IT'S JUST A KEEP-ALIVE TIMER.
ON EXIT, -4(SP) HOLDS THE REMNANTS OF THE 800. MS COUNT.

```
WTCRDY: MOV #800, -(SP)    ; SET 800 MSEC TIMER.
1$:     BIT 3CRDY, @RLCS    ; CONTROLLER READY ??
        BNE 2$              ; YES, EXIT
        JSR R5, WDELAY      ; WAIT A WHILE
        DEC (SP)            ; APPROX A MILLISECOND
        BNE 1$              ; CHECK IF TIME UP
        JSR R5, AFTER       ; NO GO BACK
        DFERR CRTIM, ERRO   ; GET REGISTERS FOR ERROR.
                                ; CONTROLLER TIMED OUT
```

TRAP CSERDF
.WORD 100
.WORD CRTIM
.WORD ERRO

```
2$:     SKP2
        JSR R5, AFTER       ; GET REGISTERS
        TST (SP)+           ; CLEAN THE STACK...
        RTS R5              ; ...AND RETURN.
```

```

906      :RLV11 MAINTENANCE SUBROUTINE FOR CRC CALCULATIONS
907      :ROUTINE TO RETRIEVE PATTERN AND CALCULATE CRC OF PATTERN+3
908      :AND CRC OF CRC OF PATTERN+4.
909      :PATTERN IS SAVED IN 'INIDA' FOR SUBSEQUENT LOADING FOR DA.
910      :CRC OF PATTERN+3 WILL BE STORED IN 'GDCRC3'.
911      :CRC OF CRC OF PATTERN+4 WILL BE STORED IN 'GDCRC4'.
912
913      :CALL: JSR      R5,CALCRC
914      :      .WORD    N      ;PATTERN FOR INITIAL DA
915
916      CALCRC: MOV      (R5)+,INIDA      ;STORE PATTERN
917      MOV      INIDA,TEMP1
918      MOV      TEMP1,TEMP5
919      ADD      #3,TEMP5      ;ADD 3 TO PATTERN
920      MOV      TEMP5,TEMP1
921      MOV      TEMP1,1$
922      JSR      R5,SIMBCC      ;CALCULATE EXPECTED CRC
923      1$:      .WORD    0      ;DATA BITS
924      .WORD    0      ;INITIAL PATTERN+3
925      MOV      CALBCC,GDCRC3      ;SAVE CRC OF PATTERN+3
926      INC      TEMP5      ;VALUE=PATTERN+4
927      MOV      TEMP5,TEMP1
928      MOV      TEMP1,2$
929      JSR      R5,SIMBCC      ;CALCULATE EXPECTED CRC
930      2$:      .WORD    0      ;DATA BITS
931      .WORD    0      ;INITIAL PATTERN+4
932      MOV      CALBCC,3$      ;STARTING CRC=0
933      JSR      R5,SIMBCC      ;STORE CRC FOR NEXT CALL
934      3$:      .WORD    0      ;CAL. CRC OF CRC OF DA+4
935      .WORD    0      ;DATA BITS
936      MOV      CALBCC,GDCRC4      ;CRC OF DA+4
937      INC      TEMP5      ;STARTING CRC=0
938      JSR      R5,SIMBCC      ;SAVE CRC OF CRC OF DA+4
939      4$:      .WORD    0
940      .WORD    0
941      MOV      CALBCC,GDCRC4
942      RTS      R5
943
944      :SUBROUTINE TO CALCULATE A CRC.
945      :ROUTINE WILL CALCULATE A CRC-16 CRC ON A WORD OF
946      :1-16 BITS IN LENGTH, RESULT IS RETURNED IN 'CALBCC'
947
948      :CALL: JSR      R5,SIMBCC
949      :      .WORD    NUMBER OF BITS (1-16)
950      :      .WORD    DATA FOR CRC CALCULATION
951      :      .WORD    PREVIOUS OR STARTING CRC
952      :      (SHOULD BE ZEROED FOR START)
953
954      SIMBCC: MOV      R0,-(SP)      ;SAVE R0
955      MOV      R1,-(SP)      ;SAVE R1
956      MOV      R2,-(SP)      ;SAVE R2
957      MOV      (R5)+,TEMP2      ;GET NUMBER OF BITS
958      MOV      (R5)+,TEMP3      ;GET DATA FOR CRC CALCULATION
959      MOV      (R5)+,TEMP4      ;GET STARTING CRC
960      1$:      CLR      BCCFBK
961      MOV      TEMP4,R0      ;GET PRESENT CRC
962      ROR      TEMP3      ;ROTATE NEW DATA
963      ADC      R0      ;MERGE NEW WITH OLD

```

962	011120	032700	(00001		BIT	#1,R0	;BIT 0 SET
963	011124	001402			BEG	2\$	;IF NOT CONTINUE
964	011126	005137	003406		COM	BCCFBK	
965	011132	013700	003404	2\$:	MOV	XPOLY,R0	;GET CRC POLYNOMIAL (CRC-16)
966	011136	005100			COM	R0	;COMPLIMENT POLYNOMIAL
967	011140	040037	003406		BIC	R0,BCCFBK	
968	011144	000241			CLC		;CLEAR CARRY
969	011146	006037	003416		ROR	TEMP4	
970	011152	013700	003406		MOV	BCCFBK,R0	
971	011156	013701	003416		MOV	TEMP4,R1	
972	011162	010102			MOV	R1,R2	
973	011164	040100			BIC	R1,R0	
974	011166	043702	003406		BIC	BCCFBK,R2	
975	011172	050200			BIS	R2,R0	
976	011174	043737	003404	003416	BIC	XPOLY,TEMP4	
977	011202	050037	003416		BIS	R0,TEMP4	
978	011206	005347	003412		DEC	TEMP2	
979	011212	001333			BNE	1\$	
980	011214	013737	003416	003410	MOV	TEMP4,CALBCC	
981	011222	012602			MOV	(SP)+,R2	
982	011224	012601			MOV	(SP)+,R1	
983	011226	012600			MOV	(SP)+,R0	
984	011230	000205			RTS	R5	;RETURN

```

986 .SBTTL MEMORY SIZER
987 *****
988
989 ROUTINE TO SIZE AVAILABLE MEMORY.
990 FREELY ADAPTED FROM H-SSIZE IN SYSMAC.SML (C3).
991 THIS ROUTINE MUST RESIDE WITHIN THE FIRST 24K (0-137776).
992
993 USE KT IF IT'S AVAILABLE, OTHERWISE SIZE UP TO 30K IN THE
994 TRADITIONAL FASHION. RETURN WITH:
995
996 MSIZE: .WORD 0 ; TOTAL MEMORY SIZE (K WORDS).
997 LSTPG: .WORD 0 ; PAGE ADDRESS (PAF) OF LAST 1K PAGE...
998 ;...OR ZERO IF KT NOT AVAILABLE.
999 LSTAD: .WORD 0 ; LAST VIRTUAL ADDRESS IN LAST PAGE...
1000 ;...OR LAST ADDRESS UNDER 30K (IF NO KT).
1001 ABUSW: .WORD 16. ; ADDRESS BUS WIDTH, 16, 18, OR 22.
1002
1003 ALL GENERAL REGISTERS ARE USED BUT NOT SAVED.
1004 MEMORY PARITY ERRORS (IF ANY) ARE IGNORED.
1005
1006 MMR0= 177572 ; KT CONTROL REGISTERS.
1007 MMR1= 177574
1008 MMR2= 177576
1009 MMR3= 172516
1010 KIPAR0= 172340 ; KERNAL, 1 SPACE, PAR 0.
1011 KIPDR0= KIPAR0-40 ; PDR 0.
1012
1013 *****
1014
1015 011242 012737 000020 011240 .SIZE: MOV #16, ABUSW ; ASSUME 16 BIT ADDRESSING.
1016 011250 013746 000004 MOV @#4, -(SP) ; SAVE BUS-ERROR...
1017 011254 013746 000006 MOV @#6, -(SP)
1018 011260 013746 000114 MOV @#14, -(SP) ;...AND PARITY VECTORS.
1019 011264 013746 000116 MOV @#16, -(SP)
1020 011270 010605 MOV SP, R5 ; SAVE STACK POINTER IN R5.
1021
1022 011272 012737 000114 MOV #116, @#114 ; IGNORE PARITY ERRORS.
1023 011300 012737 000002 MOV #RTI, @#116
1024 011306 012737 011332 000004 MOV #18, @#4 ; SIZE USING '18' IF KT ISN'T THERE.
1025 011314 012737 000340 000006 MOV #340, @#6
1026 011322 005737 177572 TST MMR0
1027 011326 000240 240
1028 011330 000417 BR .SIZKT ; OTHERWISE, SIZE USING THE KT.
1029
1030 011332 012737 011360 000004 1$: MOV #38, @#4 ; NO KT -- SET TRAP CATCHER.
1031 011340 005002 CLR R2 ; FIRST ADDRESS (0).
1032 011342 005003 CLR R3
1033 011344 005712 2$: TST (R2) ; SIZE FROM 0 UP...
1034 011346 000240 240
1035 011350 062702 004000 ADD #4000, R2 ;...IN 1K STEPS...
1036 011354 005203 INC R3
1037 011356 000772 BR 2$
1038 011362 162702 000002 3$: SUB #2, R2 ;...UNTIL WE TRAP.
1039 011364 005001 CLR R1 ; R2 = LAST VIRTUAL ADDRESS.
1040 011366 000506 BR .SIZXIT ; R1 = 0 (PAF DOESN'T APPLY).
1041 ; RETURN.

```

1042	011370	012701	172340		.SIZKT: MOV	#KIPAR0,R1	: 151 'PAR' ADDRESS...
1043	011374	012702	172300		MOV	#KIPDR0,R2	: ...AND IT'S 'PDR'...
1044	011400	012703	000010		MOV	#8,R3	: ...AND THERE ARE 8 OF EACH.
1045	011404	005000			CLR	R0	: 1ST PAGE IS ZERO.
1046	011406	010021		1\$:	MOV	R0,(R1)+	: SET PAR'S = 0, 4K, 8K ... 28K.
1047	011410	012722	077406		MOV	#77406,(R2)+	: SET PDR'S = 4K, EX-UP, READ/WRITE.
1048	011414	062700	000200		ADD	#200,R0	: SET NEXT PAGE PAF (+4K)...
1049	011420	077306			SUB	R3,1\$	: ...AND LOOP UNTIL ALL LOADED.
1050	011422	012741	177600		MOV	#177600,-(R1)	: PAR7 IS THE I/O PAGE (PAF).
1051	011426	005041			CLR	-(R1)	: PAR6 WILL DO THE SIZING.
1052	011430	005003			CLR	R3	: R3 WILL COUNT THE K'S.
1053							
1054	011432	012737	011504	000004	MOV	#2\$,2#4	: TRAP TO 2\$ IF NO 22 BIT SUPPORT.
1055	011440	005737	172516		TST	#4R3	: 22 BITS SUPPORTED ??
1056	011444	012737	000020	172516	MOV	#20,MMR3	: MUST BE, SET 22 BIT MODE...
1057	011452	012737	011520	000004	MOV	#3\$,2#4	: ...TRAP TO 3\$ IF 22 BIT ADDRESS IS NXM.
1058	011460	005237	177572		INC	MMR0	: ***** KT ON *****
1059	011464	012711	010000		MOV	#10000,(R1)	: SET PAR6 AT START OF 22 BIT LAND...
1060	011470	023737	000200	140200	CMP	#200,140200	: ...AND LOOK FOR WRAP-AROUND.
1061	011476	001010			BNE	3\$	: WE'RE HERE IF IT DIDN'T TRAP...
1062							: ...BRANCH IF IT DIDN'T WRAP-AROUND.
1063	011500	005037	172516		CLM	MMR3	: WRAP-AROUND -- MUST BE 18 BITS ONLY.
1064	011504	012737	000022	011240	2\$: MOV	#18,.ABUSW	: SET THE BUS WIDTH = 18.
1065	011512	012704	007600		MOV	#7600,R4	: SET SIZER LIMIT = 124K.
1066	011516	000405			BR	4\$	
1067	011520	012737	000026	011240	3\$: MOV	#2,.ABUSW	: WE HAVE A REAL 22 BIT ADDRESS SPACE.
1068	011526	012704	170000		MOV	#170000,R4	: SET SIZER LIMIT = 1920K.
1069	011532				4\$:		
1070	011532	012737	011560	000004	.SIZMEM: MOV	#2\$,2#4	: TRAP TO 2\$ WHEN DONE SIZING.
1071	011540	005011			CLR	(R1)	: SET PAR6 AT 1ST PAGE (0).
1072	011542	005737	140000		1\$: TST	140000	: SIZE USING PAR6 (+0)...
1073	011546	062711	000040		ADD	#40,(R1)	: ...IN 1K STEPS.
1074	011552	005203			INC	R3	
1075	011554	021104			CMP	(R1),R4	: REACHED LIMIT ??
1076	011556	103771			BLO	1\$	: LOOP IF NOT.
1077	011560	011100		2\$: MOV	(R1),R0		: DONE, SAVE FINAL PAR6...
1078	011562	012711	001400		MOV	#1400,(R1)	: ...AND RESET IT TO BANK 6 (24K).
1079	011566	005037	177572		CLR	MMR0	: ***** KT OFF *****
1080	011572	010001			MOV	R0,R1	: RECOVER SIZING RESULT.
1081	011574	162701	000040		SUB	#40,R1	: R1 = LAST 1K PAGE (PAF).
1082	011600	012702	003776		MOV	#3776,R2	: R2 = LAST ADDR IN THAT PAGE.
1083							
1084	011604	010137	011234		.SIZXIT: MOV	R1,.LSTPG	: RETURN LAST PAGE (PAF)...
1085	011610	010237	011236		MOV	R2,.LSTAD	: ...LAST VIRTUAL ADDRESS...
1086	011614	010337	011232		MOV	R3,.MSIZE	: ...AND TOTAL MEMORY SIZE (K).
1087	011620	010506			MOV	R5,SP	: RECOVER OUR STACK POINTER...
1088	011622	012637	000116		MOV	(SP)+,2#116	: ...AND THE ERROR VECTORS.
1089	011626	012637	000114		MOV	(SP)+,2#114	
1090	011632	012637	000006		MOV	(SP)+,2#6	
1091	011636	012637	000004		MOV	(SP)+,2#4	
1092	011642	000207			RTS	PC	: ...AND RETURN.

MACY11 30C(1063) 08-FEB-82 10:07 ^{H 4} PAGE 8-9
REPORT ENVIRONMENT

1094			
1095			
1096			
1097			
1098			
1099	011644		
(3)	011644		
1100	011644	105037	012104
1101	011650	005737	011234
1102	011654	001003	
1103	011656	112737	000040
1104	011664		
(8)	011664	012746	012104
(7)	011670	012746	011764
(6)	011674	012746	000002
(3)	011700	010600	
(4)	011702	104417	
(4)	011704	062706	000006
1105	011710		
(8)	011710	013746	011232
(7)	011714	012746	012014
(6)	011720	012746	000002
(3)	011724	010600	
(4)	011726	104417	
(4)	011730	062706	000006
1106	011734		
(8)	011734	013746	011240
(7)	011740	012746	012047
(6)	011744	012746	000002
(3)	011750	010600	
(4)	011752	104417	
(4)	011754	062706	000006
1107	011760		
(4)	011760	000167	
(3)	011762	000126	
1108	011764	047045	040445
1109	012014	047045	040445
1110	012047	045	022516
1111	012104	047040	052117
1112		012112	
1113	012112		
(3)	012112		
(3)	012112	104425	

```

; SUBROUTINE TO DISPLAY THE CURRENT OPERATING ENVIRONMENT.
; CALLED ON EVERY START/RESTART, OR DIRECTLY VIA 'ZRI'.

```

```

1$:      CLRB      MSGNOT
        TST       LSTPG
        BNE       2$
        MOVB      #40,MSGNOT
2$:      PRINTF    #MSG2,#MSGNOT      : MMU

```

```
PRINTF #MSG3,.MSIZE      ; MEM SIZE.
```

```
38: PRINTF #MSG4,.ABUSW : ADDRESS SPACE.
```

EXIT RPT

```
MSG2:      .ASCIIZ  'XNXA MMUXTXA AVAILABLE'
MSG3:      .ASCIIZ  'XNXA MEMORY SIZE XD4XA KW'
MSG4:      .ASCIIZ  'XNXA XD2XA BIT ADDRESSINGXN'
MSGNOT:    .ASCIIZ  'NOT'
           .EVEN
           ENDRPT
```

```
MOV    #MSGNOT, -(SP)
MOV    #MSG2, -(SP)
MOV    #2, -(SP)
MOV    SP, R0
TRAP   CS$PNTF
ADD    #6, SP
```

```
MOV    .MSIZE,-(SP)
MOV    #MSG3,-(SP)
MOV    #2,-(SP)
MOV    SP,R0
TRAP   CS$PNTF
ADD    #6,SP
```

```
MOV    .ABUSW,-(SP)
MOV    #MSG4,-(SP)
MOV    #2,-(SP)
MOV    SP,R0
TRAP   CS$PNTF
ADD    #6,SP
```

```
.WORD JSJMP
.WORD L10012-2-
```

L10012: TRAP CSRPT


```
1115 .SBTTL
1116 .SBTTL RL DISKLESS CONTROLLER TESTS.
1117 .SBTTL
1118 .SBTTL 1 -- RLCS ADDRESSABILITY.
1119
1120 012114 STARS
1121 (2) :*****
1122 1121 :TEST TO SEE IF WE CAN ADDRESS THE CONTROL
1123 1122 :AND STATUS REGISTER. IF WE TRAP WE WILL REPORT
1124 1123 :THE ERROR AND ABORT. AFTER THIS TEST WE ONLY KNOW
1125 1124 :THAT WE CAN ADDRESS THE REGISTER.
1126 012114 STARS
1127 (2) :*****
1128 1126 012114 BEGIN.TEST
1129 (4) 012114
1130 1127 012114 005037 010626 18: CLR TRPFLG ;CLEAR TRAP OCCURANCE T1::
1131 1128 012120 012777 177777 171176 MOV #177777, @RLCS ; WRITE RLCS.
1132 1129 012126 000240 000240 240,240
1133 1130 012132 005777 171166 TST @RLCS ; READ RLCS.
1134 1131 012136 000240 000240 240,240
1135 1132 012142 005737 010626 TST TRPFLG ; EITHER ONE TRAP ??
1136 1133 012146 001413 BEQ 3$ ; NO, PROCEED.
1137 1134 012150 013737 003324 003432 MOV RLCS, GDDAT ;SET UP ERROR DATA
1138 1135 012156 104454 SFERR EM1, ERR1 ;BUS TIMEOUT IN ADDRESSING RLCS
1139 (5) 012156 104454 TRAP CSERSF
1140 (6) 012160 000002 .WORD 2
1141 (6) 012162 030706 .WORD EM1
1142 (6) 012164 026170 .WORD ERR1
1143 1136 012166 DODU UNITST ; DROP...
1144 (3) 012166 013700 003322 MOV UNITST, RC
1145 (3) 012172 104451 TRAP CSDODU
1146 1137 012174 DOCLN ;...AND ABORT.
1147 (3) 012174 104444 TRAP CSDCLN
1148 1138 012176 3$: ENDTST L10013:
1149 (3) 012176 TRAP CSETST
1150 (3) 012176 104401
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 06-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 J 4
2 -- RLBA ADDRESSABILITY. PAGE 10

SEQ 0048

```
1140 .SBTTL 2 -- RLBA ADDRESSABILITY.
1141
1142 012200 STARS
1143 (2) :*****
1144 1143 :TEST TO SEE IF WE CAN ADDRESS THE BUS ADDRESS
1145 1144 :REGISTER. IF WE TRAP WE WILL REPORT THE ERROR
1146 1145 :AND ABORT. AFTER THIS TEST WE ONLY KNOW THAT
1147 1146 :WE CAN ADDRESS THE REGISTER.
1148 012200 STARS
1149 (2) :*****
1150 (4) 012200 BEGIN.TEST
1151 012200 005037 010626 171114 1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE T2::
1152 012200 012777 177777 ;WRITE RLBA.
1153 012200 000240 000240 240,240
1154 012200 005777 171104 TST @RLBA ; READ RLBA.
1155 012200 000240 000240 240,240
1156 012200 005737 010626 TST TRPFLG ; ANY TRAPS ??
1157 012200 001413 003326 003432 BEQ 3$ ;NO, CONTINUE
1158 012200 013737 003326 003432 MOV RLBA,GDDAT ;SETUP ERROR DATA
1159 (5) 012200 104454 SFERR EM2,ERR1 ;BUS TIMEOUT IN ADDRESSING RLBA
1160 (6) 012200 0000C3 TRAP CSERSF
1161 (6) 012200 030733 .WORD 3
1162 (6) 012200 026170 .WORD EM2
1163 012200 DODU UNITST ; DROP... .WORD ERR1
1164 (3) 012200 013700 003322 MOV UNITST,R0
1165 (3) 012200 104451 TRAP CSODDU
1166 012200 DOCLN ;...AND ABORT. TRAP CSODCLN
1167 (3) 012200 104444 3$: ENDTST L10014: TRAP CSETST
1168 (3) 012200 104401
```


CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 K 4
3 -- RLDA ADDRESSABILITY. PAGE 11

SEQ 0049

1162
1163
1164 012264

.SBTTL 3 -- RLDA ADDRESSABILITY.

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

STARS

BEGIN.TEST

T3::

1170 012264
(4) 012264
1171 012264 005037 010626
1172 012270 012777 177777 171032
1173 012276 000240 0C0240
1174 012302 005777 171022
1175 012306 000240 000240
1176 012312 005737 010626
1177 012316 001413
1178 012320 013737 003330 003432
1179 012326

1\$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
MOV #177777,@RLDA ;WRITE RLDA
240,240
TST @RLDA ;READ RLDA
240,240
TST TRPFLG ;ANY TRAPS ??
BEQ 3\$;NO, CONTINUE
MOV RLDA,GDDAT ;SETUP ERROR INFO
SFERR EM3,ERR1 ;BUS TIMEOUT IN ADDRESSING RLDA

TRAP CSERSF
.WORD 4
.WORD EM3
.WORD ERR1

(5) 012326 104454
(6) 012330 000004
(6) 012332 030760
(6) 012334 026170
1180 012336
(3) 012336 013700 003322
(3) 012342 104/51

DODU UNITST ;DROP...

MOV UNITST,R0
TRAP CSDODU

1181 012344 104444
(3) 012344
1182 012346
(3) 012346
(3) 012346 104401

3\$: ENDTST

...AND ABORT.

TRAP CSDCLN

L10015:

TRAP CSETST

```
1184 .SBTTL 4 -- RLMP ADDRESSABILITY.
1185
1186 012350 STARS
1187 (2) *****
1188 :TEST TO SEE IF WE CAN ADDRESS THE MULTIPURPOSE
1189 :REGISTER. IF WE TRAP WE WILL REPORT THE ERROR AND
1190 :ABORT. AFTER THIS TEST WE ONLY KNOW THAT WE CAN
1191 :ADDRESS THE REGISTER.
1192 STARS
1193 (2) *****
1194 (4) BEGIN TEST
1195 012350 005037 010626 170750 1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE T4::
1196 012354 012777 177777 ;WRITE RLMP
1197 012362 000240 000240 240,240 ; READ RLMP
1198 012366 005777 170740 ; ANY TRAPS ??
1199 012372 000240 000240 240,240 ;NO CONTINUE
1200 012376 005737 010626 TST TRPFLG ;SET UP ERROR INFO
1201 012402 001413 BEQ 3$ ;BUS TIMEOUT IN ADDRESSING RLMP
1202 012412 013737 003332 003432 MOV RLMP,GDDAT
1203 (5) 012412 104454 SFERR EM4,ERR1
1204 (6) 012414 000005 TRAP CSERSF
1205 (6) 012416 031005 .WORD S
1206 (6) 012420 026170 .WORD EM4
1207 012422 013700 003322 DODU UNITST ; DROP... .WORD ERR1
1208 (3) 012422 104451 MOV UNITST,RO
1209 (3) 012426 104444 TRAP CSODDU
1210 012430 104444 DOCLN ;...AND ABORT. TRAP CSODCLN
1211 (3) 012432 3$: ENDTST L10016: TRAP CSETST
1212 (3) 012432 104401
```

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 M 4
5 -- RLBAE ADDRESSABILITY (RLV12 ONLY).

SEQ 0051

```
1206 .SBTTL 5 -- RLBAE ADDRESSABILITY (RLV12 ONLY).
1207
1208 012434 STARS
1209 (2) ;*****
1210 012434 ; THE DEFAULT (SPIP) CONFIGURATION REQUIRES THAT THE
1211 (2) ; BAE REGISTER BE ENABLED. CHECK IT AND REPORT ACCORDINGLY.
1212 012434 STARS
1213 (4) ;*****
1214 012434 BEGIN.TEST
1215 012434 023727 003402 000002 CMP RLTP,#RLV12 ; RLV12 ??
1216 012442 103444 4$ ; NO, SKIP THIS TEST.
1217 012444 005037 010626 1$: CLR TRPFLG ; CLEAR TRAP OCCURANCE
1218 012450 013737 003334 003432 MOV RLBAE,GDDAT ; SET ADDRESS IN CASE OF ERROR.
1219 012456 012777 177777 170650 MOV #-1,RLBAE ; WRITE BAE.
1220 012464 000240 000240 240.240 TST @RLBAE ; READ BAE.
1221 012470 005777 170640 240.240 TST @RLBAE ; READ BAE.
1222 012474 000240 000240 240.240 TST @RLBAE ; READ BAE.
1223 012500 023727 003402 000003 CMP RLTP,#RLV12X ; DEFAULT CONFIG (W/BAE) ??
1224 012506 001013 3$ BNE 3$ ; BR IF NOT.
1225 012510 005737 010626 2$: TST TRPFLG ; BAE SHOULD HAVE ANSWERED -- DID IT ??
1226 012514 001417 4$ BEQ 4$ ; BR IF SO.
1227 012516 104454 SFERR EM4A,ERR1 ; BUS TIMEOUT IN ADDRESSING RLBAE
1228 (5) 012516 104454 TRAP CSERSF
1229 (6) 012520 000006 .WORD 6
1230 (6) 012522 031032 .WORD EM4A
1231 (6) 012524 026170 .WORD ERR1
1232 012526 013700 003322 DODJ UNITST ; DROP...
1233 (3) 012526 104451 MOV UNITST,R0
1234 (3) 012532 104451 TRAP CSDODU
1235 012534 104444 DOCLN ;...AND ABORT.
1236 (3) 012534 104444 TRAP CSDCLN
1237 012536 005737 010626 3$: TST TRPFLG ; BAE SHOULD HAVE TRAPPED -- DID IT ??
1238 012542 001004 4$ BNE 4$ ; BR IF SO.
1239 012544 104455 DFERR EM4B,ERR1 ; BAE IS NOT DISABLED.
1240 (5) 012544 104455 TRAP CSERDF
1241 (6) 012546 000145 .WORD 101
1242 (6) 012550 03106C .WORD EM4B
1243 (6) 012552 026170 .WORD ERR1
1244 012554 4$: ENDTST
1245 (3) 012554 104401 L10017: TRAP CSETST
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 N 4
6 -- BUS RESET OF ALL REGISTERS.

SEQ 0052

1235
1236
1237 012556

.SBTTL 6 -- BUS RESET OF ALL REGISTERS.

STARS

TEST THAT A BUS RESET WILL CLEAR THE PROPER BITS.

IN THE CONTROL AND STATUS REGISTER, THOSE BITS ARE

1-6, 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.

THE OTHER REGISTERS SHOULD GO TO ZERO.

STARS

BEGIN TEST

T6::

SETPRI #PRI07

;PRIORITY TO SEVEN

MOV #PRI07,RO
TRAP CSSPRI

1249 012556 012700 000340

1250 012556 104441

1251 012556 012777 000377 170532

1252 012556 012737 000200 003532

1253 012556 032777 040000 170516

1254 012556 001403

1255 012556 052737 140000 003532

1256 012556 012777 177777 170502

1257 012556 012777 177777 170476

1258 012556 023737 003402 000003

1259 012556 001003

1260 012556 012777 177777 170464

1261 012556 004537 010160

1262 012556 000372

1263 012556 000005

1264 012556 004537 010160

1265 012556 000372

1266 012556 017737 170432 003434

1267 012556 042737 000001 003434

1268 012556 023737 003434 003432

1269 012556 001404

1270 012556 012712 104455

1271 012556 000146

1272 012556 033311

1273 012556 026220

1274 012556 005037 003432

1275 012556 017737 170374 003434

1276 012556 001404

1277 012556 104455

1278 012556 000146

1279 012556 033311

1280 012556 026220

1281 012556 017737 170356 003434

1282 012556 001404

1283 012556 104455

1284 012556 000146

1285 012556 033311

1286 012556 026220

1287 012556 017737 170356 003434

1288 012556 001404

1289 012556 104455

MOV #377,@RLCS ;LOAD ALL RLCS LOADABLE BITS
MOV #CRDY,GDCAT ;SETUP EXPECTED CSR.
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 #-1,@RLBA ;LOAD ALL BITS IN THE OTHERS.
MOV #-1,@RLDA
CMP RLTP,#RLV12X ;RLV12X ??
BNE 2\$;NO, SKIP NEXT.
MOV #-1,@RLBAE
JSR R5,WDELAY ;DELAY BEFORE...
250.
RESET
JSR R5,WDELAY ;...AND AFTER.
250.
MOV @RLCS,BDDAT ;READ RLCS
BIC #DRDY,BDDAT ;CLEAR OUT DRDY - DON'T CARE
CMP BDDAT,GDDAT ;DID INIT WORK ON CSR.
BEQ 4\$;YES, BRANCH
DFERR EM69,ERR2 ;BUS-INIT FAILED ON CSR.

TRAP CSERDF
.WORD 103
.WORD EM69
.WORD ERR2

4\$: CLR GDDAT ;EXPECT 0 IN THE OTHERS.

MOV @RLBA,BDDAT
BEQ 5\$
DFERR EM70,ERR2 ;BUS-INIT FAILED ON RLBA.

TRAP CSERDF
.WORD 103
.WORD EM70
.WORD ERR2

5\$: MOV @RLDA,BDDAT
BEQ 6\$
DFERR EM71,ERR2 ;BUS-INIT FAILED ON RLDA.

TRAP CSERDF

B 5

SEQ 0053

```
.WORD 104
.WORD EM71
.WORD ERR2
```

```
TRAP      CSERDF
.WORD     105
.WORD     EM71A
.WORD     ERR2
```

L10020: TRAP CSETST

```

1284
1285
1286 013020
1287
1288
1289
1290
1291 013020
1292
1293 013020
1294 013020 012703 004100
1295 013024 104404
1296 013026 011337 003432
1297 013028 052737 000200 003432
1298 013030 013777 003432 170256
1299 013032 032777 040000 170250
1300 013034 001403
1301 013036 052737 140000 003432
1302 013038 017737 170234 003434
1303 013040 042737 000001 003434
1304 013042 023737 003432 003434
1305 013044 001404
1306 013110 104455
1307 013112 000152
1308 013114 031127
1309 013116 026220
1310 013120 104410
1311 013122 000012
1312 013124 005723
1313 013126 020327 004176
1314 013128 001335
1315 013130
1316 013132 104405
1317 013134
1318 013136 104401

```

```

.SBTTL 7 -- READ WRITE OF RLCS.

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
*****
      BEGIN.TEST
      MOV      #CSPAT,R3          ;SET UP TABLE POINTER OF PATTERNS
      BGNSEG                      ;****START OF SEGMENT****
      1$:      MOV      (R3),GDDAT ;GET PATTERN INTO GDDAT
      BIS      #200,GDDAT         ;INSURE GO IS SET
      MOV      GDDAT,@RLCS        ;LOAD RLCS (CONTROL AND STATUS)
      BIT      #DERR,@RLCS        ;IF DRIVE ERROR PRESENT
      BEQ      2$                 ;THEN EXPECT DRIVE AND
      BIS      #ERR!DERR,GDDAT    ;COMPOSITE ERROR
      2$:      MOV      @RLCS,BDDAT ;READ RLCS BACK
      BIC      #DRDY,BDDAT        ;IGNORE DRIVE READY
      CMP      GDDAT,BDDAT        ;DID WE READ WHAT WE LOADED
      BEQ      3$                 ;YES, THEN BRANCH
      DFERR    EM5,ERR2           ;WRONG DATA IN RLCS
      TRAP     CSBSEG

      3$:      ESCAPE SEG          ; ESCAPE AND LOOP (IF FLA:LOE).
      TRAP     C$ERDF
      .WORD    106
      .WORD    EM5
      .WORD    ERR2
      TRAP     C$ESCAPE
      .WORD    10000$-.

      TST      (R3)+              ;BUMP FOR NEXT PATTERN
      CMP      R3,#CSEND          ;CHECK FOR END
      BNE      1$                 ;NOT END, LOAD NEXT PATTERN
      ENDSEG                      ;****END OF SEGMENT****
      10000$:
      TRAP     C$ESEG

      L10021:
      TRAP     C$ETST

```



```

1313 .SBTTL      8 -- READ WRITE OF RLBA.
1314
1315 013140
1316 (2)
1317 STARS
1318 :*****
1319 :TEST THAT WE CAN WRITE/READ BITS 15 THRU 1 OF THE
1320 :BUS ADDRESS REGISTER.  FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
1321 :GROWING 0 AND SHIFTING 0.  BIT 0 IS ALSO LOADED BUT
1322 :SHOULD ALWAYS COME BACK AS 0
1323 STARS
1324 :*****
1325 BEGIN.TEST
1326
1327      T8::
1328      MOV      #BEGPAT,R3      ;GET START OF PATTERN LIST
1329      BGNSEG      ;****START OF SEGMENT****
1330
1331      1$:      MOV      (R3),GDDAT      ;GET PATTERN TO SEND
1332      CMP      RL11,RL11      ;RL11 ??
1333      BNE      2$      ;NO.
1334      BIC      #BIT0,GDDAT      ;YES, KEEP RLBA EVEN (UNIBUS)
1335      MOV      GDDAT,@RLBA      ;LOAD PATTERN TO BUS ADDRESS
1336      MOV      @RLBA,BDDAT      ;READ IT BACK
1337      CMP      GDDAT,BDDAT      ;IS IT CORRECT?
1338      BEQ      3$      ;IF SO, BRANCH
1339      DFERH      EM6,ERR2      ;DATA WRONG IN RLBA
1340
1341      2$:      TRAP      C$BSEG
1342
1343      3$:      ESCAPE      SEG      ; ESCAPE AND LOOP (IF FLA;LOE).
1344
1345      TST      (R3)+      ;BUMP FOR NEXT PATTERN
1346      CMP      R3,#ENDPAT      ;CHECK FOR END
1347      BNE      1$      ; LOOP TIL DONE.
1348      ENDSEG      ;****END OF SEGMENT****
1349
1350      10000$:      TRAP      C$ESEG
1351
1352      L10022:      TRAP      C$SETST
1353
1354      ENDTST
1355
1356      004076
1357
1358      104405
1359
1360      104401

```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 E 5
9 -- READ WRITE OF RLDA. PAGE 17

SEQ 0056

```
1340 .SBTTL 9 -- READ WRITE OF RLDA.
1341
1342 013244 STARS
1343 (2) :*****
1344 1343 :TEST THAT WE CAN WRITE/READ THE DISK ADDRESS REGISTER
1345 1344 :ALL BIT POSITIONS ARE WRITTEN USING FOUR PATTERNS:
1346 013244 :GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
1347 (2) STARS
1348 013244 :*****
1349 013250 BEGIN.TEST
1350 (3) 013250 104404 T9::
1351 013252 011337 003432 ;SET UP POINTER TO PATTERN LIST
1352 013254 013777 003432 170044 ;****START OF SEGMENT****
1353 013272 023737 003432 003434 TRAP CSBSEG
1354 013300 001404 ;GET PATTERN
1355 013302 104455 ;LOAD PATTERN IN DA
1356 (5) 013302 000154 ;READ PATTERN BACK
1357 (6) 013304 031226 ;IS IT CORRECT?
1358 (6) 013306 026220 ;BRANCH IF CORRECT
1359 (6) 013310 004076 ;WRONG DATA IN RLDA
1360 013312 2$: ESCAPE SEG ; ESCAPE AND LOOP (IF FLA:LOE).
1361 (3) 013312 000012 TRAP C$ERDF
1362 (3) 013314 005723 .WORD 108
1363 013316 020327 004076 .WORD EM7
1364 013320 001352 .WORD ERR2
1365 013322 001352 TST (R3)+ ;BUMP POINTER
1366 (3) 013324 001352 CMP R3,#ENDPAT ;AT END OF PATTERNS?
1367 (3) 013326 001352 BNE 1$ ;NO, BRANCH BACK
1368 013326 10000$: TRAP C$ESEG
1369 (3) 013326 104405 ;****END OF SEGMENT****
1370 (3) 013330 104401 ENDTST L10023: TRAP C$ETST
```


CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 F 5 PAGE 18
10 -- READ WRITE OF RLBAE (RLV12 ONLY).

SEQ 0057

```
1363 .SBTTL 10 -- READ WRITE OF RLBAE (RLV12 ONLY).
1364
1365 013332 STARS
1366 (2) :*****
1367 :TEST THAT WE CAN WRITE/READ THE BUS ADDRESS EXTENSION REGISTER.
1368 :ALL BIT POSITIONS ARE WRITTEN USING FOUR PATTERNS:
1369 :GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
1370 013332 :ONLY THE LOW 6 BITS ARE TESTED RLBAE<5:0>
1371 (2) STARS
1372 (4) :*****
1373 013332 BEGIN.TEST
1374 013340 023727 003402 000003 CMP RLTP,#RLV12X ; BAE ENABLED ?? T10::
1375 013346 001035 BNE 3$ ; NO, DON'T EVEN TRY.
1376 (3) 013346 012703 003670 MOV #BEGPAT,R3 ;SET UP POINTER TO PATTERN LIST
1377 013350 104404 BGNSEG ;****START OF SEGMENT**** TRAP C$BSEG
1378 013354 011337 003432 1$: MOV (R3),GDDAT ;GET PATTERN
1379 013362 013777 003432 167752 MOV GDDAT, @RLBAE ;LOAD PATTERN IN BAE
1380 013370 017737 167746 003434 MOV @RLBAE,BDDAT ;READ PATTERN BACK
1381 013376 042737 177700 003432 BIC #^C77,GDDAT ; ** ONLY 5:0 ARE VALID BITS.
1382 013406 023737 003432 003434 CMP GDDAT,BDDAT ;IS IT CORRECT?
1383 (5) 013406 104455 BEQ 2$ ;BRANCH IF CORRECT
1384 (6) 013410 000155 DFERR EM8,ERR2 ;WRONG DATA IN RLBAE TRAP C$ERDF
1385 (6) 013412 031254 .WORD 109
1386 (6) 013414 026220 .WORD EM8
1387 013416 104410 2$: ESCAPE SEG ; ESCAPE AND LOOP (IF FLA;LOE). .WORD ERR2
1388 (3) 013416 000012 TRAP C$ESCAPE
1389 (3) 013420 005723 .WORD 10000$-.
1390 013422 005723 TST (R3)+ ;BUMP POINTER
1391 013424 020327 004076 CMP R3,#ENDPAT ;AT END OF PATTERNS?
1392 013430 001347 BNE 1$ ;NO, BRANCH BACK
1393 (3) 013432 ENDSEG ;****END OF SEGMENT****
1394 (3) 013432 104405 10000$: TRAP C$ESEG
1395 013434 104401 3$: ENDTST L10024: TRAP C$ETST
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 PAGE 19
11 -- BIS AND BIC OF RLCS.

SEQ 0058

```
1390 .SBTTL 11 -- BIS AND BIC OF RLCS.
1391
1392 013436
1393 (2)
1394
1395 STARS
1396 :*****
1397 :TEST THAT WE CAN USE THE 'BIS' AND 'BIC' INSTRUCTIONS ON THE CONTROL
1398 :AND STATUS REGISTER. BITS 8,9 AND 6-1 ARE TESTED TO
1399 :SET INDIVIDUALLY AS WELL AS COLLECTIVELY WITHOUT DESTROYING
1400 :ANY PREVIOUS DATA PATTERN
1401 STARS
1402 :*****
1403 BEGIN.TEST
1404
1405 T11::
1406 MOV #CSPAT,R3 ;GET BEGINNING OF LIST
1407 BGNSEG ;**** START SEGMENT ****
1408
1409 012703 004100 104404 TRAP C$BSEG
1410 (3) 013442 012777 000200 167652 1$: MOV #CRDY,@RLCS ;INSURE GO IS THERE
1411 013442 011337 003432 MOV (R3),GDDAT ;SET UP EXPECTED RLCS...
1412 013442 052737 000200 003432 BIS #CRDY,GDDAT ;...IN GDDAT
1413 013442 051377 167634 BIS (R3),@RLCS ;*** BIT SET PATTERN IN RLCS
1414 013470 032777 040000 167626 BIT #DERR,@RLCS ;IF ERROR BIT SET THEN...
1415 013476 001403 BEQ 2$ ;...EXPECT IT ON THE READ BACK.
1416 013500 052737 140000 003432 BIS #ERR!DERR,GDDAT
1417 013506 017737 167612 003434 2$: MOV @RLCS,BDDAT ;READ RLCS TO CHECK 'BIS'
1418 013514 042737 000001 003434 BIC #DRDY,BDDAT ;CLEAR OUT DRIVE READY
1419 013522 023737 003434 003432 CMP BDDAT,GDDAT ;DID BIS WORK?
1420 013530 001404 BEQ 3$ ;BRANCH IF OKAY
1421 013532 104455 DFERR EM61,ERR2 ;WRONG DATA IN RLCS ON BIS.
1422 (5) 013532 000156 TRAP C$ERDF
1423 (6) 013534 032617 .WORD 110
1424 (6) 013536 026220 .WORD EM61
1425 013540 .WORD ERR2
1426 013542 104410 3$: ESCAPE SEG ; ESCAPE AND LOOP (IF FLA;LOE).
1427 (3) 013542 000114 TRAP C$ESCAPE
1428 (3) 013544 000114 .WORD 10000$-.
1429 013546 012777 001776 167550 4$: MOV #1776,@RLCS ;SET ALL SETTABLE BITS
1430 013554 012737 001776 003432 MOV #1776,GDDAT ;SET UP EXPECT DATA IN...
1431 013562 041337 003432 BIC (R3),GDDAT ;...GDDAT
1432 013566 041377 167532 BIC (R3),@RLCS ;*** XCT BIC ON RLCS.
1433 013572 032777 040000 167524 BIT #DERR,@RLCS ;IF DRIVE ERROR BIT SET...
1434 013600 001403 BEQ 5$ ;...EXPECT IT SET WHEN WE READ.
1435 013602 052737 140000 003432 BIS #ERR!DERR,GDDAT
1436 013610 017737 167510 003434 5$: MOV @RLCS,BDDAT ;MOVE RLCS TO BDDAT FOR COMPARE
1437 013616 042737 000001 003434 BIC #DRDY,BDDAT ;CLEAR DRIVE READY
1438 013624 023737 003434 003432 CMP BDDAT,GDDAT ;DID 'BIC' WORK PROPERLY
1439 013632 001404 BEQ 6$ ;BRANCH IF OKAY
1440 013634 104455 DFERR EM62,ERR2 ;WRONG DATA IN RLCS ON BIC.
1441 (5) 013634 000157 TRAP C$ERDF
1442 (6) 013636 032665 .WORD 111
1443 (6) 013640 026220 .WORD EM62
1444 013642 .WORD ERR2
1445 013644 104410 6$: ESCAPE SEG ; ESCAPE AND LOOP (IF FLA;LOE).
1446 (3) 013644 000012 TRAP C$ESCAPE
1447 (3) 013646 005728 .WORD 10000$-.
1448 013650 020327 004176 TST (R3)+ ;GET NEXT PATTERN
1449 013652 001272 CMP R3,#CSEND ;AT END OF LIST
1450 013656 001272 BNE 1$ ;NO, GO BACK WITH NEXT PATTERN
```

CVRLBA0 -- RLV12 DISKLESS MACY11 30G(1063) 08-FEB-82 10:07 H 5
CVRLBA.P11 08-FEB-82 09:59 11 -- BIS AND BIC OF RLCS. PAGE 19-1

SEQ 0059

1430 013660
(3) 013660
(3) 013660 104405
1431 013662
(3) 013662
(3) 013662 104401

ENDSEG

ENDTST

100008: TRAP CSESEG

L10025: TRAP CSETST

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 ^{1 5} PAGE 20
12 -- BIS AND BIC OF RLBA.

SEQ 0060

```
1433 .SBTTL 12 -- BIS AND BIC OF RLBA.
1434
1435 013664 STARS
1436 ;*****
1437 ;TEST THAT THE 'BIS' AND 'BIC' INSTRUCTIONS WILL WORK ON THE
1438 ;BUS ADDRESS REGISTER. BITS 15-0 ARE LOADED, ONLY BITS 15-1
1439 ;ARE EXPECTED BACK. FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
1440 ;GROWING 0, AND SHIFTING 0.
1441 STARS
1442 ;*****
1443 BEGIN.TEST
1444
1445 013664 T12::
1446 013664 MOV #BEGPAT,R3 ;GET START OF LIST
1447 013670 BGNSEG ;**** START SEGMENT ****
1448 013670 104404 TRAP C$BSEG
1449 013672 005077 167430 1$: CLR @RLBA ;CLEAR 'BA'
1450 013676 011337 003432 MOV (R3),GDDAT ;SET EXPECTED
1451 013702 023727 003402 000000 CMP RLTP,#RL11 ;RL11 ??
1452 013710 001063 BNE 2$ ;NO.
1453 013712 042737 000001 003432 BIC #1,GDDAT ;BIT 0 CAN'T SET IN RLBA (UNIBUS)
1454 013720 051377 167402 2$: BIS (R3),@RLBA ;XCT BIS RLBA WITH PATTERN
1455 013724 017737 167376 003434 MOV @RLBA,BDDAT ;READ 'BA'
1456 013732 023737 003434 003432 CMP BDDAT,GDDAT ;DID RLBA LOAD PROPERLY?
1457 013740 001404 BEQ 3$ ;BRANCH IF YES
1458 013742 104455 DFERR EM63,ERR2 ;WRONG DATA IN RLBA ON BIS.
1459 013744 000160 TRAP C$ERDF
1460 013746 032735 .WORD 112
1461 013750 026220 .WORD EM63
1462 013752 3$: ESCAPE SEG ; ESCAPE AND LOOP (IF FLA:LOE). .WORD ERR2
1463 013754 104410 TRAP C$ESCAPE
1464 013756 000070 .WORD 10000$-.
1465 013758 012777 177776 167342 4$: MOV #-2,@RLBA ;SET RLBA TO ALL 1'S (BIT 0=0)
1466 013764 012737 177776 003432 MOV #-2,GDDAT ;SET UP EXPECTED RESULTS
1467 013772 041337 003432 BIC (R3),GDDAT ;IN GDDAT
1468 013776 041377 167324 BIC (R3),@RLBA ;XCT BIC RLBA
1469 014002 017737 167320 003434 MOV @RLBA,BDDAT ;READ RLBA
1470 014010 023737 003434 003432 CMP BDDAT,GDDAT ;BIC WORK OKAY?
1471 014016 001404 BEQ 5$ ;IF YES BRANCH
1472 014020 104455 DFERR EM64,ERR2 ;WRONG DATA IN RLBA ON BIC.
1473 014022 000161 TRAP C$ERDF
1474 014024 033003 .WORD 113
1475 014026 026220 .WORD EM64
1476 014030 5$: ESCAPE SEG ; ESCAPE AND LOOP (IF FLA:LOE). .WORD ERR2
1477 014032 104410 TRAP C$ESCAPE
1478 014034 000012 .WORD 10000$-.
1479 014036 005723 004076 TST (R3)+ ;GET NEXT PATTERN
1480 014042 020327 CMP R3,#ENDPAT ;HAVE WE COMPLETED LIST
1481 014044 001313 BNE 1$ ;NO, GO BACK FOR NEXT
1482 014046 104405 ENDSEG 10000$: TRAP C$ESEG
1483 014048 104401 ENDTST L10026: TRAP C$ETST
```

```

1470 .SBTTL 13 -- BIS AND BIC OF RLDA.
1471
1472 014050 STARS
(2) :*****
1473 :TEST THAT THE 'BIS' AND 'BIC' INSTRUCTIONS WILL WORK ON THE DISK
1474 :ADDRESS REGISTER. BITS 15-0 ARE TESTED WITH 4 PATTERNS, GROWING 1,
1475 :SHIFTING 1, GROWING 0, AND SHIFTING 0.
1476 014050 STARS
(2) :*****
1477 014050 BEGIN.TEST
(4) 014050 T13::
1478 014050 012703 003670 MOV #BEGPAT,R3 ;GET START OF LIST
1479 014054 BGNSEG ; **** START SEGMENT ****
(3) 014054 104404 TRAP C$BSEG
1480 014056 005077 167246 1$: CLR @RLDA ;CLEAR 'DA'
1481 014062 011337 003432 MOV (R3),GDDAT ;SET EXPECTED
1482 014066 051377 167236 BIS (R3),@RLDA ;XCT BIS RLDA
1483 014072 017737 167232 003434 MOV @RLDA,BDDAT ;READ RLDA
1484 014100 023737 003434 003432 CMP BDDAT,GDDAT ;IS RLDA CORRECT
1485 014106 001404 BEQ 2$ ;IF OKAY BRANCH
1486 014110 DFERR EM65,ERR2 ;WRONG DATA IN RLDA ON BIS.
(5) 014110 104455 TRAP C$ERDF
(6) 014112 000162 .WORD 114
(6) 014114 033053 .WORD EM65
(6) 014116 026220 .WORD ERR2
1487 014120 2$: ESCAPE SEG ; ESCAPE AND LOOP (IF FLA;LOE).
(3) 014120 104410 TRAP C$ESCAPE
(3) 014122 002070 .WORD 10000$-.
1488 014124 012777 177777 167176 3$: MOV #-1,@RLDA ;SET RLDA TO ALL 1'S
1489 014132 012737 177777 003432 MOV #-1,GDDAT ;SET EXPECTED DATA
1490 014140 041337 003432 BIC (R3),GDDAT ;SET EXPECTED DATA
1491 014144 041377 167160 BIC (R3),@RLDA ;XCT 'BIC' RLDA
1492 014150 017737 167154 003434 MOV @RLDA,BDDAT ;READ RLDA
1493 014156 023737 003432 003434 CMP GDDAT,BDDAT ;DID 'BIC' WORK?
1494 014164 001404 BEQ 4$ ;IF IT DID BRANCH
1495 014166 DFERR EM66,ERR2 ;WRONG DATA IN RLDA ON BIC.
(5) 014166 104455 TRAP C$ERDF
(6) 014170 000163 .WORD 115
(6) 014172 033121 .WORD EM66
(6) 014174 026220 .WORD ERR2
1496 014176 4$: ESCAPE SEG ; ESCAPE AND LOOP (IF FLA;LOE).
(3) 014176 104410 TRAP C$ESCAPE
(3) 014200 000012 .WORD 10000$-.
1497 014202 005723 TST (R3)+ ;GET NEXT PATTERN
1498 014204 020327 004076 CMP R3,#ENDPAT ;DONE?
1499 014210 001322 BNE 1$ ;NO GO BACK
1500 014212 ENDSEG
(3) 014212 10000$: TRAP C$ESEG
(3) 014212 104405
1501 014214 ENDTST
(3) 014214 L10027: TRAP C$ETST
(3) 014214 104401

```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 K 5
14 -- BIS AND BIC OF RLBAE (RLV12 ONLY).

SEQ 0062

```
1503 .SBTTL 14 -- BIS AND BIC OF RLBAE (RLV12 ONLY).
1504
1505 014216 STARS
1506 (2) :*****
1507 1506 :TEST THAT THE 'BIS' AND 'BIC' INSTRUCTIONS WILL WORK ON THE
1508 1507 :BUS ADDRESS EXTENSION REGISTER. ALL BITS ARE LOADED BUT ONLY <5:0>
1509 1508 :ARE EXPECTED BACK. FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
1510 1509 :GROWING 0, AND SHIFTING 0.
1511 1510 STARS
1512 (2) :*****
1513 1511 BEGIN.TEST
1514 (4) 014216 T14::
1515 1512 023727 003402 000003 CMP RLTP,#RLV12X ;BAE ENABLED ??
1516 1513 001070 BNE 5$ ;SKIP IT IF NOT.
1517 1514 014224 C12703 003670 MCV #BEGPAT,R3 ;GET START OF LIST
1518 1515 014232 BGNSEG ;**** START SEGMENT ****
1519 (3) 014232 104404
1520 1516 014234 005077 167074 1$: CLR @RLBAE ;CLEAR 'BAE'
1521 1517 014240 011337 003432 MOV (R3),GDDAT ;SET EXPECTED...
1522 1518 014244 042737 177700 003432 BIC #^C77,GDDAT ;...ONLY <5:0> ARE VALID.
1523 1519 014252 051377 167056 BIS (R3),@RLBAE ;XCT BIC RLBAE WITH PATTERN
1524 1520 014256 017737 167052 003434 MOV @RLBAE,BDDAT ;READ 'BAE'
1525 1521 014264 023737 003434 003432 CMP BDDAT,GDDAT ;DID IT LOAD PROPERLY?
1526 1522 014272 001404 BEQ 2$ ;BRANCH IF YES
1527 1523 014274 104455 DFERR EM67,ERR2 ;WRONG DATA IN RLBAE ON BIS.
1528 (5) 014274 104455 TRAP C$ERDF
1529 (6) 014276 000164 .WORD 116
1530 (6) 014300 033171 .WORD EM67
1531 (6) 014302 026220 .WORD ERR2
1532 1524 014304 104410 2$: ESCAPE SEG ;ESCAPE AND LOOP (IF FLA;LOE).
1533 (3) 014304 104410 TRAP C$ESCAPE
1534 (3) 014306 000076 .WORD 10000$-.
1535 1525 014310 012777 177777 167016 3$: MOV #-1,@RLBAE ;SET RLBAE TO ALL 1'S
1536 1526 014316 012737 177777 003432 MOV #-1,GDDAT ;SET UP EXPECTED RESULTS...
1537 1527 014324 041337 003432 BIC (R3),GDDAT ;...IN GDDAT...
1538 1528 014330 042737 177700 003432 BIC #^C77,GDDAT ;...<5:0> ONLY.
1539 1529 014336 041377 166772 BIC (R3),@RLBAE ;XCT BIC RLBAE
1540 1530 014342 017737 166766 003434 MOV @RLBAE,BDDAT ;READ RLBAE
1541 1531 014350 023737 003434 003432 CMP BDDAT,GDDAT ;BIC WORK OKAY?
1542 1532 014356 001404 BEQ 4$ ;IF YES BRANCH
1543 1533 014360 104455 DFERR EM68,ERR2 ;WRONG DATA IN RLBAE ON BIC.
1544 (5) 014360 104455 TRAP C$ERDF
1545 (6) 014362 000165 .WORD 117
1546 (6) 014364 033240 .WORD EM68
1547 (6) 014366 026220 .WORD FRR2
1548 1534 014370 104410 4$: ESCAPE SEG ;ESCAPE AND LOOP (IF FLA;LOE).
1549 (3) 014370 104410 TRAP C$ESCAPE
1550 (3) 014372 000012 .WORD 10000$-.
1551 1535 014374 005723 TST (R3)+ ;GET NEXT PATTERN
1552 1536 014376 020327 004076 CMP R3,#ENDPAT ;HAVE WE COMPLETED LIST
1553 1537 014402 001314 BNE 1$ ;NO, GO BACK FOR NEXT
1554 1538 014404 ENDSEG
1555 (3) 014404 104405 10000$: TRAP C$ESEG
1556 1539 014406 5$: ENDTST L10030:
```


CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 L⁵ PAGE 22-1
14 -- BIS AND BIC OF RLBAE (RLV12 ONLY).

(3) 014406 104401

TRAP CSETST

SEQ 0063

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 M 5
15 -- UNIQUENESS OF RLCS. PAGE 23

SEQ 0064

1541
1542
1543 014410

.SBTTL 15 -- UNIQUENESS OF RLCS.

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.
: IF RLV12, CHECK THE BAE ALSO, NOTING THAT CSR<5:4> SHOULD
: BE MIRRORED IN BAE<1:0>.

STARS

BEGIN.TEST

1552 014410
(4) 014410

1553 014400 012777 177776 166710
1554 014406 012777 177777 166704
1555 014404 023727 003402 000003
1556 014432 001003

MOV #-2, @RLBA ;SET RLBA TO ALL 1'S
MOV #-1, @RLDA ;SET RLDA TO ALL 1'S
CMP RLTP, #RLV12X
BNE 1\$

1557 014414 012777 000077 166672
1558 014402 012777 000200 166654
1559 014450 012737 177776 003432
1560 014456 017737 166644 003434
1561 014454 023737 003432 003434
1562 014472 001405

1\$: MOV #77, @RLBAE ; SET BAE TO ALL 1'S.
MOV #CRDY, @RLCS ;WRITE RLCS
MOV #-2, GDDAT ;SET UP EXPECTED BA.
MOV @RLBA, BDDAT ;READ RLBA
CMP GDDAT, BDDAT
BEQ 2\$; BR IF BA UNCHANGED.
DFERR EM72, ERR2 ;CS MODIFIED BA

1563 014474 104455
(5) 014474 000166
(6) 014476 033472
(6) 014500 026220

TRAP CSERDF
.WORD 118
.WORD EM72
.WORD ERR2

1564 014504 104406
(3) 014504 012737 177777 003432
1565 014506 017737 166610 003434
1566 014514 023737 003432 003434
1567 014502 001405
1568 014500 104455
1569 014502 000167

CKLOOP
2\$: MOV #-1, GDDAT ;SET UP EXPECTED DA.
MOV @RLDA, BDDAT ;READ DA
CMP GDDAT, BDDAT
BEQ 3\$; BR IF DA UNCHANGED.
DFERR EM73, ERR2 ;CS MODIFIED DA

TRAP CSCLP1

(5) 014532 104455
(6) 014534 000167
(6) 014536 033525
(6) 014540 026220

TRAP CSERDF
.WORD 119
.WORD EM73
.WORD ERR2

1570 014542 104406
(3) 014542 023727 003402 000003
1571 014544 001016
1572 014552 012737 000074 003432
1573 014554 017737 166546 003434
1574 014562 023737 003432 003434
1575 014570 001404
1576 014576 104455
1577 014600 000170

CKLOOP
3\$: CMP RLTP, #RLV12X
BNE 4\$
MOV #74, GDDAT ; SET EXPECTED BAE.
MOV @RLBAE, BDDAT ; READ IT.
CMP GDDAT, BDDAT
BEQ 4\$; BR IF BAE IS RIGHT.
DFERR EM73A, ERR2 ; BAE WRONG AFTER WRITING CS.

TRAP CSCLP1

(5) 014600 104455
(6) 014602 000170
(6) 014604 033560
(6) 014606 026220

TRAP CSERDF
.WORD 120
.WORD EM73A
.WORD ERR2

1578 014610
(3) 014610

4\$: ENDTST

L10031:

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59 MACY11 306(1063) 08-FEB-82 10:07 N⁵ PAGE 23-1
15 -- UNIQUENESS OF RLCS.

(3) 014610 104401

TRAP CSETST

SEQ 0065

```

1580 .SBTTL 16 -- UNIQUENESS OF RLBA.
1581
1582 014612 STARS
1583 (2) :*****
1584 :TEST THE UNIQUENESS OF THE BUS ADDRESS REGISTER. THE
1585 :RLCS AND RLDA ARE LOADED WITH XXX20X AND 177777
1586 :RESPECTIVELY. THE RLBA IS THEN WRITTEN TO INSURE
1587 :THAT NEITHER THE RLCS OR RLDA ARE MODIFIED.
1588 014612 : IF RLV12, CHECK THAT BAE IS UNAFFECTED ALSO.
1589 (2) STARS
1590 (4) :*****
1591 014612 BEGIN.TEST
1592 014620 012737 000200 003432 MOV #CRDY,GDDAT ;CONTROLLER READY
1593 014626 032777 040000 166476 BIT #DERR,@RLCS ;IF DRIVE ENROR IS SET...
1594 014630 001403 BEQ 1$
1595 014636 052737 140000 003432 BIS #ERR!DERR,GDDAT ;...EXPECT IT BACK.
1596 014644 013777 003432 166460 1$: MOV GDDAT,@RLCS ;LOAD RLCS
1597 014652 012777 177777 166456 MOV #-1,@RLDA ;LOAD RLDA
1598 014660 023727 003402 000003 CMP RLTP,#RLV12X
1599 014662 001003 BNE 2$
1600 014670 012777 000074 166444 2$: MOV #74,@RLBAE ;LOAD RLBAE.
1601 014674 005077 166432 CLR @RLBA ;WRITE TO RLBA
1602 014702 017737 166424 003434 MOV @RLCS,BDDAT ;READ RLCS
1603 014710 042737 000001 003434 BIC #DRDY,BDDAT ;IGNORE DRIVE READY
1604 014716 023737 003432 003434 CMP GDDAT,BDDAT
1605 014720 001405 BEQ 3$ ;BR IF CS UNCHANGED.
1606 014722 104455 DFERR EM74,ERR2 ;BA MODIFIED CS
1607 014724 000171 TRAP C$ERDF
1608 014726 033617 .WORD 121
1609 014728 026220 .WORD EM74
1610 014730 .WORD ERR2
1611 014732 104406 CKLOOP
1612 014734 012737 177777 003432 3$: MOV #-1,GDDAT ;SET EXPECTED DA.
1613 014740 017737 166364 003434 MOV @RLDA,BDDAT ;READ IT.
1614 014746 023737 003432 003434 CMP GDDAT,BDDAT
1615 014754 001405 BEQ 4$ ;BR IF DA UNCHANGED.
1616 014756 104455 DFERR EM75,ERR2 ;BA MODIFIED DA
1617 014758 000172 TRAP C$ERDF
1618 014760 033651 .WORD 122
1619 014762 026220 .WORD EM75
1620 014764 .WORD ERR2
1621 014766 104406 CKLOOP
1622 014768 023727 003402 000003 4$: CMP RLTP,#RLV12X
1623 014770 001016 BNE 5$
1624 015000 012737 000074 003432 MOV #74,GDDAT ;SET EXPECTED BAE.
1625 015006 017737 166322 003434 MOV @RLBAE,BDDAT ;READ IT.
1626 015014 023737 003432 003434 CMP GDDAT,BDDAT
1627 015022 001404 BEQ 5$ ;BR IF BAE UNCHANGED.
1628 015024 104455 DFERR EM75A,ERR2 ;BA MODIFIED BAE
1629 015026 000173 TRAP C$ERDF
1630 015030 033703 .WORD 123
1631 015032 026220 .WORD EM75A
1632 .WORD ERR2

```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07^{C 6} PAGE 24-1
16 -- UNIQUENESS OF RLBA.

SEQ 0067

1619 015034
(3) 015034
(3) 015034 104401

58: ENDTST

L10032: TRAP CSETST

CVRLBAO -- RLV12 DISKLESS,
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 ^{D 6} PAGE 25
17 -- UNIQUENESS OF RLDA.

SEQ 0068

```
1621 .SBTTL 17 -- UNIQUENESS OF RLDA.
1622
1623 015036 STARS
1624 (2) :*****
1625 :TEST THE UNIQUENESS OF THE DISK ADDRESS REGISTER. THE RLCS
1626 :AND RLBA ARE LOADED WITH XXX20X AND 177776 RESPECTIVELY.
1627 :THE RLDA IS THEN WRITTEN TO INSURE THAT
1628 :NEITHER THE RLCS OR THE RLBA ARE MODIFIED.
1629 :IF RLV12, CHECK THAT RLBAE IS UNAFFECTED ALSO.
1630 015036 STARS
1631 (2) :*****
1632 015036 BEGIN.TEST
1633 (4) 015036 T17::
1634 015036 012737 000200 003432 MOV #CRDY,GDDAT ;CONTROLLER READY
1635 015044 032777 040000 166252 BIT #DERR,RLCS ;IF DRIVE ERROR IS SET...
1636 015052 001403 BEQ 1$
1637 015054 052737 140000 003432 BIS #ERR!DERR,GDDAT ;...EXPECT IT BACK.
1638 015062 013777 003432 166252 1$: MOV GDDAT,RLCS ;LOAD CS
1639 015070 012777 177776 166250 MOV #-2,RLBA ;LOAD BA WITH ALL 1'S
1640 015076 023727 003402 000003 CMP RLTP,RLV12X
1641 015104 001003 BNE 2$
1642 015106 012777 000074 166220 MOV #74,RLBAE ; LOAD BAE.
1643 015114 005077 166210 2$: CLR RLDA ; WRITE TO RLDA.
1644 015120 017737 166200 003434 MOV RLCS,BDDAT ;READ RLCS
1645 015126 042737 000001 003434 BIC #DRDY,BDDAT ;IGNORE DRIVE READY
1646 015134 023737 003432 003434 CMP GDDAT,BDDAT
1647 015142 001405 BEQ 3$ ; BR IF CS UNCHANGED.
1648 015144 104455 DFERR EM76,ERR2 ;DA MODIFIED CS
1649 (5) 015144 104455 TRAP CSERDF
1650 (6) 015146 000174 .WORD 124
1651 (6) 015150 033737 .WORD EM76
1652 (6) 015152 026220 .WORD ERR2
1653 015154 104406 CKLOOP
1654 (3) 015154 104406 TRAP CSCLP1
1655 015156 012737 177776 003432 3$: MOV #-2,GDDAT ; SET EXPECTED BA.
1656 015164 017737 166136 003434 MOV RLBA,BDDAT ; READ IT.
1657 015172 023737 003432 003434 CMP GDDAT,BDDAT
1658 015200 001405 BEQ 4$ ; BR IF BA UNCHANGED.
1659 015202 104455 DFERR EM77,ERR2 ;DA MODIFIED BA
1660 (5) 015202 104455 TRAP CSERDF
1661 (6) 015204 000175 .WORD 125
1662 (6) 015206 033772 .WORD EM77
1663 (6) 015210 026220 .WORD ERR2
1664 015212 104406 CKLOOP
1665 (3) 015212 104406 TRAP CS'LP1
1666 015214 023727 003402 000003 4$: CMP RLTP,RLV12X
1667 015222 001016 BNE 5$
1668 015224 012737 000074 003432 MOV #74,GDDAT ; SET EXPECTED BAE.
1669 015232 017737 166076 003434 MOV RLBAE,BDDAT ; READ IT.
1670 015240 023737 003432 003434 CMP GDDAT,BDDAT
1671 015246 001404 BEQ 5$ ; BR IF BAE UNCHANGED.
1672 015250 104455 DFERR EM77A,ERR2 ; DA MODIFIED BAE.
1673 (5) 015250 104455 TRAP CSERDF
1674 (6) 015252 000176 .WORD 126
1675 (6) 015254 034025 .WORD EM77A
1676 (6) 015256 026220 .WORD ERR2
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07^{E 6} PAGE 25-1
17 -- UNIQUENESS OF RLDA.

SEQ 0069

1660 015260
(3) 015260
(3) 015260 104401

SS: ENDTST

L10033: TRAP CSETST

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 F 6
18 -- UNIQUENESS OF RLMP.

SEQ 0070

```
1662 .SBTTL 18 -- UNIQUENESS OF RLMP.
1663
1664 015262 STARS
1665 (2) :*****
1666 :TEST THE UNIQUENESS OF THE MULTI-PURPOSE REGISTER
1667 :WE WILL WRITE THE RLCS, RLBA, AND THE RLDA, THEN THE
1668 :RLMP IS WRITTEN. WE THEN GO BACK AND VERIFY THE CONTENTS
1669 :OF THE RLCS, RLBA, RLDA.
1670 015262 : IF RLV12, INCLUDE THE RLBAE ALSO.
1671 (2) STARS
1672 (4) :*****
1673 015262 BEGIN.TEST
1674 015270 012737 000200 003432 MOV #CRDY,GDDAT ;CONTROLLER READY T18::
1675 015276 032777 040000 166026 BIT #DERR,RLCS ;IF DRIVE ERROR IS SET...
1676 015300 001403 BEQ 1$ ;
1677 015306 052737 140000 003432 BIS #ERR!DERR,GDDAT ;...EXPECT IT BACK LATER.
1678 015314 012777 003432 166010 1$: MOV GDDAT,RLCS ;LOAD CS
1679 015322 012777 177776 166004 MOV #-2,RLBA ;LOAD BA WITH ALL 1'S
1680 015330 012777 177777 166000 MOV #-1,RLDA ;LOAD RLDA
1681 015336 023727 003402 000003 CMP RLTP,RLV12X
1682 015340 001003 BNE 2$ ;
1683 015346 012777 000074 165766 2$: MOV #74,RLBAE ;LOAD RLBAE.
1684 015352 005077 165760 CLR @RLMP ;WRITE TO RLMP
1685 015360 017737 165746 003434 MOV @RLCS,BDDAT ;READ RLCS
1686 015366 042737 000001 003434 BIC #DRDY,BDDAT ;IGNORE DRIVE READY
1687 015374 023737 003432 003434 CMP GDDAT,BDDAT
1688 015376 001405 BEQ 3$ ; BR IF CS OK.
1689 (5) 015376 104455 DFERR EM44,ERR2 ;MP MODIFIED CS
1690 (6) 015400 000177 TRAP CSERDF
1691 (6) 015402 032333 .WORD 127
1692 (6) 015404 026220 .WORD EM44
1693 015406 104406 .WORD ERR2
1694 (3) 015406 104406 CKLOOP TRAP CSCLP1
1695 015410 012737 177776 003432 3$: MOV #-2,GDDAT ; SET EXPECTED BA.
1696 015416 017737 165704 003434 MOV @RLBA,BDDAT ; READ IT.
1697 015424 023737 003432 003434 CMP GDDAT,BDDAT
1698 015432 001405 BEQ 4$ ; BR IF BA OK.
1699 015434 104455 DFERR EM45,ERR2 ;MP MODIFIED BA
1700 (5) 015434 104455 TRAP CSERDF
1701 (6) 015436 000200 .WORD 128
1702 (6) 015440 032366 .WORD EM45
1703 (6) 015442 026220 .WORD ERR2
1704 015444 104406 CKLOOP TRAP CSCLP1
1705 (3) 015444 104406
1706 015446 012737 177777 003432 4$: MOV #-1,GDDAT ; SET EXPECTED DA.
1707 015454 017737 165650 003434 MOV @RLDA,BDDAT ; READ IT.
1708 015462 023737 003432 003434 CMP GDDAT,BDDAT
1709 015470 001405 BEQ 5$ ; BR IF DA OK.
1710 015472 104455 DFERR EM46,ERR2 ;MP MODIFIED DA
1711 (5) 015472 104455 TRAP CSERDF
1712 (6) 015474 000201 .WORD 129
1713 (6) 015476 032421 .WORD EM46
1714 (6) 015500 026220 .WORD ERR2
1715 015502 CKLOOP
```


SEQ 0071

Address	Hex	Assembly	Comments	Symbol
1701	015502	104406		
(3)	015502	003402	000003	5\$: CMP RLTP,#RLV12X
1702	015512	001016		
(3)	015512	000074	003432	BNE 6\$
1703	015514	012737		
(3)	015514	000074	003432	MOV #74,GDDAT ; SET EXPECTED BAE.
1704	015522	017737		
(3)	015522	165606	003434	MOV @RLBAE,BDDAT ; READ IT.
1705	015530	023737	003432	003434
(3)	015530	003432	003434	CMP GDDAT,6DDAT
1706	015536	001404		
(3)	015536	001404		BEG 6\$; BR IF BAE OK.
1707	015540			
(3)	015540			DFERR EM47,ERR2 ; MP MODIFIED BAE.
(5)	015540	104455		
(6)	015542	000202		
(6)	015544	032454		
(6)	015546	026220		
1708	015550			
(3)	015550		6\$:	ENDTST
(3)	015550			
(3)	015550	104401		

TRAP C\$CLP1

TRAP C\$ERDF

.WORD 130

.WORD EM47

.WORD ERR2

L10034:

TRAP C\$ETST

SEQ 0072

Address	Offset	Instruction	Comment	Trap	Value
1710		.SBTTL 19 -- UNIQUENESS OF RLBAE (RLV12 ONLY).			
1711					
1712	015552	STARS			
1713	(2)	*****			
1714		:TEST THAT WRITING TO THE RLBAE HAS NO AFFECT ON			
1715		:THE RLBA AND RLDA REGISTERS. THE RLCS REGISTER WILL			
1716		:BE AFFECTED, BUT ONLY IN THE EXTENDED ADDRESS BITS <5:4>.			
1717	015552	STARS			
1718	(2)	*****			
1719	015552	BEGIN.TEST			
1720	(4)				
1721	015552	023727 003402 000003	CMP RLTP,#RLV12X ; RLV12 WITH BAE ??		
1722	015552	001071	BNE 4\$; EXIT IF NOT.		
1723	015552	012737 000200 003432	MOV #CRDY,GDDAT ; CONTROLLER READY		
1724	015552	032777 040000 165526	BIT #DERR,@RLCS ; IF DRIVE ERROR IS SET...		
1725	015552	001403	BEQ 1\$		
1726	015552	052737 140000 003432	BIS #ERR!DERR,GDDAT ; ...EXPECT IT BACK LATER.		
1727	015552	013777 003432 165510	MOV GDDAT,@RLCS ; LOAD CS, NOTE THAT <5:4> = 0		
1728	015552	005077 165506	CLR @RLBA ; 0 => BA		
1729	015552	005077 165504	CLR @RLDA ; 0 => DA		
1730	015552	012777 000077 165502	MOV #77,@RLBAE ; WRITE TO RLBAE.		
1731	015552	052737 000060 003432	BIS #60,GDDAT ; SET EXPECTED CS.		
1732	015552	017737 165460 003434	MOV @RLCS,BDDAT ; READ IT.		
1733	015552	042737 000001 003434	BIC #DRDY,BDDAT ; IGNORE DRIVE READY BIT.		
1734	015552	023737 003432 003434	CMP GDDAT,BDDAT		
1735	015552	001405	BEQ 2\$; BR IF CS IS RIGHT.		
1736	015552	104455	DFERR EM50,ERR2 ; CS WRONG AFTER WRITING BAE.	TRAP	C\$ERDF
1737	015552	000203		.WORD	131
1738	015552	032510		.WORD	EM50
1739	015552	026220		.WORD	ERR2
1740	015552		CKLOOP	TRAP	C\$CLP1
1741	015552	104406			
1742	015552	005037 003432	CLR GDDAT ; EXPECT ZERO ON THE REST.		
1743	015552	017737 165420 003434	MOV @RLBA,BDDAT ; READ BA		
1744	015552	001405	BEQ 3\$; BR IF UNAFFECTED.		
1745	015552		DFERR EM51,ERR2 ; BAE MODIFIED BA.	TRAP	C\$ERDF
1746	015552	104455		.WORD	132
1747	015552	000204		.WORD	EM51
1748	015552	032547		.WORD	ERR2
1749	015552	026220			
1750	015552		CKLOOP	TRAP	C\$CLP1
1751	015552	104406			
1752	015552	017737 165400 003434	MOV @RLDA,BDDAT ; READ DA.		
1753	015552	001404	BEQ 4\$; BR IF UNAFFECTED.		
1754	015552		DFERR EM52,ERR2 ; BAE MODIFIED DA.	TRAP	C\$ERDF
1755	015552	104455		.WORD	133
1756	015552	000205		.WORD	EM52
1757	015552	032573		.WORD	ERR2
1758	015552	026220			
1759	015552		4\$: ENDTST		
1760	015552	104401			
1761	015552			L10035:	
1762	015552			TRAP	C\$ETST

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 PAGE 28
20 -- FUNCTION CODE 0, NOP (RL11), OR MAINT (RLV11/12).

SEQ 0073

1746
1747
1748 015746

.SBTTL 20 -- FUNCTION CODE 0, NOP (RL11), OR MAINT (RLV11/12).

STARS

: TEST FUNCTION CODE 0.
: IF RL11 -- NOP. EXPECT CS<13:10> CLEAR, OTHER REGISTERS UNAFFECTED.
: IF RLVXX -- MAINT. SETUP IN NORMAL FASHION, BUT ONLY CHECK FOR
: FUNCTION COMPLETE (NO ERRORS AND DA = INITIAL DA+6).
: WE'LL CHECK THE REST OF THE MAINT FUNCTION LATER.

STARS

: BEGIN.TEST

T20::

: INITIAL SETUP ASSUMES THAT WE HAVE AN RLV.
: IF NOT -- NOT TO WORRY -- IT SHOULD JUST NO-OP.
: ALL WE CARE ABOUT IS THAT WE GET NO ERRORS (OR HUNG RL).

1761 015746 004537 010714
1762 015752 000000
1763 015754 004537 010210
1764 015760 000000 177001
1765 015764 004537 010640
1766 015770 023727 003402 000000
1767 015776 001047

1\$: JSR R5,CALCRC ; MAKE A CRC EVEN THO WE WON'T CHECK IT.
: .WORD 0 ; THIS WILL BE THE INITIAL DA WORD.
: JSR R5,LDFUN ; LOAD UP THE FUNCTION.
: 0, -511. ; FUNCTION 0, WC -511.
: JSR R5,WTCRDY ; WAIT FOR CONTROLLER.
: CMP RL1YP,#RL11 ; NOW ARE WE RL11 ??
: BNE 6\$; CHECK 'MAINT' FINAL STATE IF NOT.

: RL11 -- CHECK FOR ERRORS AFTER THE NO-OP.

1771 016000 005037 003356
1772 016004 013737 003350 003432
1773 016012 042737 036000 003432
1774 016020 013737 003364 003434
1775 016026 023737 003432 003434
1776 016034 001405
1777 016036

2\$: CLR B.MP ; MPR WILL BE 0.
: MOV B,CS,GDDAT ; EXPECT 'NO-OP' SHOULD HAVE
: BIC #036000,GDDAT ; ...CLEARED ANY ERROR BITS <13:10>...
: MOV E,CS,BDDAT ; ...THAT MIGHT HAVE BEEN ON.
: CMP GDDAT,BDDAT ; IS THAT WHAT WE HAVE ??
: BEQ 3\$; YES.
: DFERR EM14,ERR2 ; CS WRONG AFTER NOP.

TRAP CSERDF
.WORD 134
.WORD EM14
.WORD ERR2

1778 016046 104406
(3) 016046 104406
1779 016050 023737 003366 003352 3\$:
1780 016056 001010
1781 016060 023737 003370 003354
1782 016066 001004
1783 016070 023737 003372 003354
1784 016076 001405
1785 016100

CKLOOP
3\$: CMP E.BA,B.BA ; BA UNALTERED ??
: BNE 4\$
: CMP E.DA,B.DA ; DA UNALTERED ??
: BNE 4\$
: CMP E.MP,B.MP ; MP UNALTERED ??
: BEQ 5\$; EXIT IF ALL THREE WERE OK.
: DFERR EM14A,ERR0 ; REGISTERS ALTERED BY NOP.

TRAP CSCLP1

(5) 016100 104455
(6) 016102 000207
(6) 016104 031440
(6) 016106 026164
1786 016110 104406
(3) 016110 104406
1787 016112
(3) 016112 104432

4\$:
5\$: EXIT TST ; RL NOP TEST DONE.
TRAP CSERDF
.WORD 135
.WORD EM14A
.WORD ERR0
TRAP CSCLP1
TRAP CSEXIT

CVRLBAO -- RLV12 DISKLESS.
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 J 6 PAGE 28-1
20 -- FUNCTION CODE 0, NOP (RL11), ON MAINT (RLV11/12).

SEQ 0074

```
(3) 016114 000042 .WORD L10036-.
1788
1789 ; RLV11 OR 12 -- CHECK SEQUENCER STATE AFTER INTERNAL DIAGNOSTIC.
1790
1791 016116 013737 003370 003434 6$: MOV E,DA,BDDAT ; GET FINAL SEQUENCER STATE.
1792 016124 023727 003434 000006 CMP BDDAT,#6 ; FINAL STATE RIGHT ??
1793 016132 001004 BNE ; NO ERROR.
1794 016134 032737 002000 003364 BIT #OPI,E,CS ; YES, OPI SHOULD BE CLEAR.
1795 016142 001405 BEQ 8$ ; IF IT IS, WE'RE FAT !!
1796 016144 7$: DFERR EM14B,ERR3 ; MAINT FAILURE, DA = SEQUENCER STATE.
(5) 016144 104455 TRAP C$ERDF
(6) 016146 000210 .WORD 136
(6) 016150 031502 .WORD EM14B
(6) 016152 026254 .WORD ERR3
1797 016154 CKLOOP
(3) 016154 104406 TRAP C$CLP1
1798 016156 8$: ENR IST
(3) 016156 L10036: TRAP C$ETST
(3) 016156 104401
```

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 K 6
21 -- TEST INTERRUPT ON FUNCTION (0) COMPLETE.

SEQ 0075

1800
1801
1802 016160
(2)
1803
1804
1805
1806
1807
1808
1809 016160
(2)
1810 016160
(4) 016160
1811 016160 005037 010634
1812 016164 012700 000000
(3) 016164 104441
(3) 016170 004537 010714
1813 016172 000000
1814 016176 004537 010210
1815 016200 000100 177001
1816 016204 004537 010640
1817 016210 005737 010634
1818 016214 001006
1819 016220 005037 003424
1820 016222 104455
1821 016226 000211
(5) 016230 031544
(6) 016232 026370
(6) 016234
1822 016236 012700 000340
(3) 016236 104441
(3) 016242
1823 016244
(3) 016244
(3) 016244 104401

.SBTTL 21 -- TEST INTERRUPT ON FUNCTION (0) COMPLETE.

STARS

:TEST THAT THE RL CAN INTERRUPT THE CPU.
:WE'LL SET CPU PRIORITY AT 0, AND EXECUTE NOP/MAINT, AS BEFORE.
:THE INTERRUPT SERVICE DOES NOTHING BUT SET A FLAG.
:WE'LL WAIT 800 MSEC FOR THAT FLAG, BEFORE DECLARING AN ERROR.
:WRONG VECTORS UNDER 1000 ARE TRAPPED BY THE DRS.
:WRONG VECTORS ABOVE 1000 ARE BIG TROUBLE !!!!
STARS

BEGIN.TEST

CLR INTFLG
SETPRI #PRI00

T21::
:CLEAR INTERRUPT FLAG
:SET PSW TO 0

MOV #PRI00,R0
TRAP C\$SPRI

JSR R5,CALCRC
.WORD 0
JSR R5,LDFUN
INTEN!0, -511
JSR R5,WTCDY
TST INTFLG
BNE IS
CLR TMO
DFERR EM15,ERR7

: AGAIN, CRC WON'T BE CHECKED.
: LOAD UP AND EXECUTE...
: NOP/MAINT AND INTERRUPT.
: WAIT FOR CONTROLLER READY.
: DID INTERRUPT OCCUR ??
: BR IF SO.
: CPU LEVEL FOR ERROR MSG.
: INTERRUPT NOT REC'D.

TRAP C\$ERDF
.WORD 137
.WORD EM15
.WORD ERR7

IS: SETPRI #PRI07

MOV #PRI07,R0
TRAP C\$SPRI

ENDTST

L10037:

TRAP C\$ETST

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 L 6
22 -- TEST INTERRUPT PRIORITY BR LEVEL.

SEQ 0076

```
1825 .SBTTL 22 -- TEST INTERRUPT PRIORITY BR LEVEL.
1826
1827 016246
1828 STARS
1829 *****
1830 :TEST THAT PRIORITY GIVEN IS ACTUAL PRIORITY OF CONTROLLER.
1831 :WE ALREADY KNOW THAT THE CONTROLLER CAN INTERRUPT. WE'LL START
1832 :AT CPU LEVEL 7 AND WORK DOWN 'TIL THE INTERRUPT COMES IN.
1833 :AS BEFORE, WE'RE USING THE NOP/MAINT FUNCTION.
1834 :NOTE THAT RLV11 IS FIXED AT BR4, RL11 AND RLV12 ARE PROGRAMABLE
1835 :AND DEFAULT TO BR5. LOC 'BPRIOR' IS PROPERLY SET AT INIT TIME.
1836 STARS
1837 *****
1838 BEGIN.TEST
1839
1840 T22::
1841 MOV #PRI07,BDDAT ;SET UP INITIAL OF 7
1842 MOV #7,TMP0 ;A COPY FOR ERROR MSG.
1843 MOV BPRIOR,GDDAT ;GET GIVEN PRIORITY
1844 BGNSEG ;**** START SEGMENT ****
1845
1846 1S: SETPRI BDDAT ; SET CURRENT PRIORITY... TRAP CSBSEG
1847
1848 CLR INTFLG ;...AND CLEAR INTERRUPT FLAG.
1849 JSR R5,CALCRC ;JUST LIKE BEFORE.
1850 .WORD 0
1851 JSR R5,LDLUN ;LOAD AND EXECUTE...
1852 INTEN!0, -511. ;...NOP/MAINT AND INTERRUPT.
1853 JSR R5,WTCRDY ;WAIT FOR CONTROLLER READY.
1854 CMP BDDAT,GDDAT ;SHOULD IT INTERRUPT
1855 BGE 2S ;BR IF NOT.
1856 TST INTFLG ;DID IT INTERRUPT ??
1857 BNE 4S ;PROCEED IF SO.
1858 BR 3S ;NO ERROR.
1859 2S: TST INTFLG ;DID IT INTERRUPT ??
1860 BEQ 4S ;PROCEED IF NOT.
1861 3S: DFERR EM16,ERR7 ; INTERRUPT PRIORITY FAILURE.
1862
1863 TRAP CSERDF
1864 .WORD 138
1865 .WORD EM16
1866 .WORD ERR7
1867
1868 CKLOOP
1869
1870 4S: SUB #40,BDDAT ;LOWER CPU PRIORITY...
1871 DEC TMP0 ;...TO NEXT LEVEL.
1872 BPL 1S
1873 ENDSEG
1874
1875 10000S:
1876 TRAP CSESEG
1877
1878 SETPRI #PRI07
1879
1880 MOV #PRI07,R0
1881 TRAP CSSPRI
1882
1883 L10040:
1884 TRAP CSETST
1885
1886 ENDTST
```

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 M 6
23 -- RLV11/12 MAINTENANCE, FORCED GP! (WC <> 511.)

SEQ 0077

```
1863 .SBTTL 23 -- RLV11/12 MAINTENANCE, FORCED OPI (WC <> 511.)
1864
1865 016414
1866 (2)
1867 STARS
1868 :*****
1869 :EXECUTE MAINTENANCE MODE WITH AN INCORRECT INITIAL WORD COUNT.
1870 :FIRST PASS WC < 510. AND SECOND PASS WC > 511. BOTH CASES SHOULD
1871 :FORCE OPI ERRORS. THE TEST SHOULD FORCE COMPOSITE ERROR(BIT 15),
1872 :HEADER NOT FOUND(BIT12) AND OPI(BIT 10). DRIVE ERROR IS IGNORED.
1873 STARS
1874 :*****
1875 BEGIN.TEST
1876
1877 T23::
1878 :*
1879 :* NOTE: IF CONTROLLER TYPE IS RL11, YOU HAVE END-PASS AT THIS POINT.
1880 :* ALL REMAINING TESTS ARE UNIQUE TO RLV11 AND/OR RLV12.
1881 :*
1882 1$: JSR R5,CALCRC ;DO CRC CALCULATION FOR...
1883 .WORD 0 ;...INITIAL DA OF ZERO.
1884 MOV #-509,3$ ;1ST WC < 510.
1885 MOVB #'<,EM27X ; ADJUST ERROR TEXT.
1886 MOVB #'0,EM27X+4
1887 2$: JSR R5,LDFUN ;PERFORM MAINT FUNCTION
1888 MAINT
1889 -509. ; TEST WORD COUNT.
1890 3$: JSR R5,WTCRDY
1891 MOV #ERR!HNF!OPI,RO ;EXPECT AT LEAST THESE ERRORS.
1892 BIC E.CS,RO
1893 BEQ 4$
1894 JSR R5,GETERR ; EXIT IF WE'RE OK.
1895 NOP ; OTHERWISE, GET THEM ALL...
1896 DFERR EM27,ERR10 ;...NONE IS EQUALLY BAD NEWS.
1897 ; STATUS INCORRECT, WC <> 511.
1898 TRAP C$ERDF
1899 (5) 016500 104455 .WORD 139
1900 (6) 016502 000213 .WORD EM27
1901 (6) 016504 032151 .WORD ERR10
1902 (6) 016506 026420
1903 1891 016510 CKLOOP TRAP C$CLP1
1904 (3) 016510 104406
1905 1892 016512 023727 016452 177000 4$: CMP 3$,#-512. ; 2ND PASS ??
1906 1893 016520 001412 BEQ 5$ ; WE'RE DONE IF SO.
1907 1894 016522 012737 177000 016452 MOV #-512,3$ ; OTHERWISE, SET WC > 511.
1908 1895 016530 112737 000076 032251 MOVB #'>,EM27X ; ADJUST ERROR TEXT.
1909 1896 016536 112737 000061 032255 MOVB #'1,EM27X+4
1910 1897 016544 000737 BR 2$ ; AND GO 'ROUND.
1911 1898 016546 5$: ENDTST
1912 (3) 016546 L10041:
1913 (3) 016546 104401 TRAP C$ETST
```

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 N 6
24 -- RLV11/12 MAINTENANCE, FORCED OPI INTERRUPT.

SEQ 0078

1901
1902
1903 016550
(2)
1904
1905
1906
1907 016550
(2)
1908 016550
(4) 016550
1909 016550 004537 010714
1910 016554 000000
1911 016556
(3) 016556 012700 000000
(3) 016562 104441
1912 016564 005037 010634
1913 016570 004537 010210
1914 016574 000100 177003
1915 016600 004537 010640
1916 016604 005737 010634
1917 016610 001005
1918 016612
(5) 016612 104455
(6) 016614 000214
(6) 016616 031641
(6) 016620 026164
1919 016622
(3) 016622 104406
1920 016624
(3) 016624 012700 000340
(3) 016630 104441
1921 016632
(3) 016632
(3) 016632 104401

.SBTTL 24 -- RLV11/12 MAINTENANCE, FORCED OPI INTERRUPT.

STARS

:PERFORM TEST OF INTERRUPT BY ISSUING RLV11 MAINTENANCE FUNCTION
:WITH A WORD COUNT OF -509. TO FORCE AN OPI ERROR.
:CHECK THAT INTERRUPT OCCURS, REPORT ERROR IF NOT.
STARS

BEGIN.TEST

1\$: JSR R5,CALCRC
.WORD 0
SETPRI #PRI00

T24::
;CALCULATE CRC FOR...
;...INITIAL DA OF ZERO.
;SET PRIORITY TO ZERO

MOV #PRI00,R0
TRAP C\$SPRI

CLR INTFLG
JSR R5,LDFUN
MAINT!INTEN,-509.
JSR R5,WTCRDY
TST INTFLG
BNE 4\$
DFERR EM17,ERRO

;CLEAR INT. FLAG
; INVALID WC TO FORCE OPI ERROR.
;WAIT FOR READY
; DID IT INTERRUPT ??
; YES, THAT'S ALL.
; NO INTERRUPT RECEIVED.

TRAP C\$ERDF
.WORD 140
.WORD EM17
.WORD ERRO

CKLOOP

4\$: SETPRI #PRI07

TRAP C\$CLP1

MOV #PRI07,R0
TRAP C\$SPRI

ENDTST

L10042:

TRAP C\$ETST

CVRLBAO -- RLV12 DISKLESS,
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 PAGE 34
25 -- RLV11/12 MAINTENANCE, OPI TIMING TEST.

SEQ 0079

1923
1924
1925 016634
(2)

1926
1927
1928
1929
1930
1931
1932
1933
1934

1935 01-634
(2)

1936 016634
(4)

1937 016634 005737 003400

1938 016640 100447

1939 016642 004537 010714

1940 016646 000000

1941 016650 012704 000002

1942 016654 004537 010210

1943 016660 000000 177003

1944 016664 004537 010640

1945 016670 012737 001440 003434

1946 016676 166637 177774 003434

1947 016704 032777 002000 164412

1948 016712 001005

1949 016714 104455

(5) 016716 000215

(6) 016720 032260

(6) 016722 026164

1950 016724 000415

1951 016726 023737 003434 003450 3\$:

1952 016734 003004

1953 016736 023737 003434 003446

1954 016744 002005

1955 016746 077436

1956 016750

(5) 016750 104455

(6) 016752 000216

(6) 016754 032306

(6) 016756 026450

1957 016760

(3) 016760

(3) 016760 104401

.SBTTL 25 -- RLV11/12 MAINTENANCE, OPI TIMING TEST.

STARS

:NOTE: THE SOFT TIMER EMPLOYED HERE IS CALIBRATED FOR LSI'S ONLY.
: IF THE CPU TYPE IS UNKNOWN (NOT LSI), BYPASS THIS TEST.

:PERFORM RLV11 MAINTENANCE FUNCTION (0) IN FLAG MODE.
:FORCE AN OPI TIMEOUT BY SETTING AN INVALID WORD COUNT.
:MEASURE THE TIME UNTIL THE ERROR FLAG SETS AND COMPARE THAT
:TIME AGAINST THE SPEC LIMITS (155 TO 650 MSEC).
: NOTE: SINCE THE TIMING LOOP IS SO GROSS, WE'LL GIVE IT
: A SECOND CHANCE BEFORE WE DECLARE AN ERROR.

STARS

BEGIN.TEST

1\$: TST CPU TYP ; LSI CPU ???
BMI 5\$; IF NOT, DON'T EVEN TRY !!!
JSR R5,CALCRC ; MAKE CRC FOR ZERO.

2\$: MOV #2,R4
JSR R5,LDFUN ;PERFORM MAINT. FUNCTION
MAINT, -509. ; MAINT WITH INVALID WC.
JSR R5,WTCRDY ; WAIT FOR DONE.
MOV #800,BDDAT ; CALCULATE OPI TIME...
SUB -4(SP),BDDAT ;...IN BDDAT.
BIT #OPI,ARLCS ; OPI ERROR ??
BNE 3\$; YES, CHECK TIMING.
DFERR EM31,ERRO ; OPI FLAG NOT RECEIVED.

TRAP CSERDF
.WORD 141
.WORD EM31
.WORD ERRO

3\$: BR 5\$
CMP BDDAT,OPI MX
BGT 4\$
CMP BDDAT,OPI MN
BGE 5\$; OK IF TIME WITHIN LIMITS.
4\$: SOB R4,2\$; TRY ONCE MORE.
DFERR EM32,ERR11 ;OPI TIMING INCORRECT

TRAP CSERDF
.WORD 142
.WORD EM32
.WORD ERR11

5\$: ENDTST

L10043:

TRAP CSETST

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 C 7
26 -- RLV11/12 MAINTENANCE, FIFO DMA AND CRC CHECK.

SEQ 0080

1959
1960
1961 016762

.SBTTL 26 -- RLV11/12 MAINTENANCE, FIFO DMA AND CRC CHECK.

STARS

:PERFORM RLV11/12 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.
:DMA 1 -- USES 28 DATA PATTERNS IN 'PATCRC' AND 'PATDAT'.
:DMA 2 -- USES COMPLIMENT OF THOSE DATA PATTERNS.
:DMA 3 -- USES A RANDOM 256 WORD PATTERN.

STARS

:BEGIN.TEST

T26::

: SUBTEST 1 -- 28 DIFFERENT DATA PATTERNS.

DMA1: BGNSUB

T26.1:

TRAP C\$BSUB

1978 016762
(3) 016762 104402
(3) 016762 012704 004200
1979 016764 016437 000072 017002
1980 016770 004537 010356
1981 016776 000000
1982 017002 012437 017014
1983 017004 004537 010714
1984 017010 000000
1985 017014 004737 017132
1986 017016 005714
1987 017022 001361
1988 017024
1989 017026
(3) 017026
(3) 017026 104403

1\$: MOV #PATCRC,R4 ;SET TABLE POINTER.
MOV 72(R4),2\$; GET TEST DATA...
JSR R5,SETPAT ;...AND FILL THE BUFFER.
2\$: .WORD 0
MOV (R4)+,3\$; GET DA TEST WORD...
JSR R5,CALCRC ;...AND CALCULATE CRC'S.
3\$: .WORD 0
JSR PC,EXECUT ; XCT MAINT MODE AND CHECK RESULTS.
TST (R4) ; LAST PATTERN DONE ??
BNE 1\$; LOOP IF NOT.
ENDSUB

L10045:

TRAP C\$ESUB

: SUBTEST 2 -- THE COMPLIMENT OF THE ABOVE 28 PATTERNS.

DMA2: BGNSUB

T26.2:

TRAP C\$BSUB

1993 017030
(3) 017030
(3) 017030 104402
1994 017032 012704 004200
1995 017036 016437 000072 017050
1996 017044 004537 010362
1997 017050 000000
1998 017052 012437 017062
1999 017056 004537 010714
2000 017062 000000
2001 017064 004737 017132
2002 017070 005714
2003 017072 001361
2004 017074
(3) 017074

1\$: MOV #PATCRC,R4 ;RESET TABLE POINTER.
MOV 72(R4),2\$; FILL WITH COMPLIMENT DATA.
JSR R5,SETCMP
2\$: .WORD 0
MOV (R4)+,3\$; CALCULATE CRC'S.
JSR R5,CALCRC
3\$: .WORD 0
JSR PC,EXECUT ; XCT
TST (R4) ; LAST ONE DONE ??
BNE 1\$; LOOP IF NOT.
ENDSUB

L10046:

CVRLBAO -- RLV12 DISKLESS.
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 D 7
26 -- RLV11/12 MAINTENANCE, FIFO DMA AND CRC CHECK.

SEQ 0081

```
(3) 017074 104403                                TRAP  C$ESUB
2005
2006      ; SUBTEST 3 -- ONE RANDOM DATA PATTERN.
2007      ;
2008      ; MA3: BGNSUB                                T26.3:
(3) 017076                                TRAP  C$BSUB
(3) 017076 104402                                ;
2009 017100 004537 010452                                ;
2010 017104 013737 003456 017116      JSR    R5,SETRAN      ;FILL BUFFER WITH RANDOM DATA.
2011 017112 004537 010714      MOV    TEMLO,1$      ; USE TEMLO FOR THE CRC WORD.
2012 017116 000000      JSR    R5,CALCRC      ; CALCULATE CRC'S.
2013 017120 004737 017132      1$:      .WORD    0
2014 017124      JSR    PC,EXECUT      ; XCT
(3) 017124      ENDSUB
(3) 017124 104403                                L10047:
2015 017126      EXIT    TST                                TRAP  C$ESUB
(3) 017126 104432                                ; ALL DONE.
(3) 017130 000530                                TRAP  C$EXIT
2016      .WORD    L10044-.
2017      ;
2018      ; THIS SUBROUTINE EXECUTES THE FUNCTION AND VERIFIES RESULTS.
2019      ; EXECUT: BGNSEG                                ; *** SET LOOP SEGMENT ***
(3) 017132 104404                                TRAP  C$BSEG
2020 017134 004537 010210      JSR    R5,LDFUN      ;PERFORM MAINT. FUNCTION
2021 017140 000000 177001      MAINT, -511.
2022 017144 004537 010640      JSR    R5,WTCRDY      ;WAIT FOR READY
2023 017150 004537 007510      JSR    R5,GETERR      ;CHECK CONTROLLER FOR ERRORS
2024 017154 000405      BR     11$      ; BR IF NONE.
2025 017156      DFERR    EM98,ERR6      ;UNEXPECTED CONTROLLER ERRORS.
(5) 017156 104455                                TRAP  C$ERDF
(6) 017160 000217                                .WORD  143
(6) 017162 034347                                .WORD  EM98
(6) 017164 026340                                .WORD  ERR6
2026 017166      CKLOOP                                TRAP  C$CLP1
(3) 017166 104406                                ;
2027 017170 032737 022000 003364 11$:      BIT    #OPI!NXM,E.CS      ; OPI OR NXM ??
2028 017176 001402      BEQ    12$      ; PROCEED IF NOT.
2029 017200      EXIT    SEG      ; YES, DON'T BOTHER WITH ANY MORE.
(3) 017202 104432                                TRAP  C$EXIT
(3) 017202 000452                                .WORD  10000$-.
2030 017204 012737 006362 003432 12$:      MOV    #BUFEND-2,GDDAT
2031 017212 013737 003366 003434      MOV    E.BA,BDDAT
2032 017220 023737 003432 003434      CMP    GDDAT,BDDAT      ;FINAL BA CORRECT ??
2033 017230 001405      BEQ    1$      ;FINAL BA INCORRECT.
2034 017230      DFERR    EM10,ERR4
(5) 017230 104455                                TRAP  C$ERDF
(6) 017232 000220                                .WORD  144
(6) 017234 031303                                .WORD  EM10
(6) 017236 026304                                .WORD  ERR4
2035 017240      CKLOOP                                TRAP  C$CLP1
(3) 017240 104406                                ;
2036 017242 013737 003354 003432 1$:      MOV    B.DA,GDDAT      ;GET BEFORE DA REGISTER
2037 017250 013737 003370 003434      MOV    E.DA,BDDAT
2038 017256 003037 003424      CLR    TMPO
2039 017262 113737 003354 003424      MOVB   B.DA,TMPO
2040 017270 062737 000006 003424      ADD    #6,TMPO      ;+6 TO DA LOW BYTE
```

2041	017276	113737	003424	003432		MOVB	TMP0,GDDAT	:STORE LOW BYTE OF DA		
2042	017304	023737	003432	003434		CMP	GDDAT,BDDAT	: FINAL DA CORRECT ??		
2043	017312	001405				BEQ	2\$			
2044	017314					DFERR	EM12,ERR4	: FINAL DA INCORRECT.		
(5)	017314	104455							TRAP	C\$ERDF
(6)	017316	000221							.WORD	145
(6)	017320	031344							.WORD	EM12
(6)	017322	026304							.WORD	ERR4
2045	017324					CKLOOP				
(3)	017324	104406							TRAP	C\$CLP1
2046	017326	013737	003442	003432	2\$:	MOV	GDCRC3,GDDAT	:GET EXPECTED CRC OF DA+3 VALUE		
2047	017334	013737	003372	003434		MOV	E.MP1,BDDAT	:GET CONTROLLER CRC OF DA+3		
2048	017342	023737	003432	003434		CMP	GDDAT,BDDAT			
2049	017350	001405				BEQ	3\$			
2050	017352					DFERR	EM20,ERR4	:CRC DA+3 INCORRECT.		
(5)	017352	104455							TRAP	C\$ERDF
(6)	017354	000222							.WORD	146
(6)	017356	031706							.WORD	EM20
(6)	017360	026304							.WORD	ERR4
2051	017362					CKLOOP				
(3)	017362	104406							TRAP	C\$CLP1
2052	017364	013737	003444	003432	3\$:	MOV	GDCRC4,GDDAT			
2053	017372	013737	003374	003434		MOV	E.MP1,BDDAT			
2054	017400	023737	003432	003434		CMP	GDDAT,BDDAT	: DITTO CRC OF CRC OF DA+4.		
2055	017406	001405				BEQ	4\$			
2056	017410					DFERR	EM21,ERR4	:CRC OF CRC OF DA+4 INCORRECT.		
(5)	017410	104455							TRAP	C\$ERDF
(6)	017412	000223							.WORD	147
(6)	017414	031757							.WORD	EM21
(6)	017416	026304							.WORD	ERR4
2057	017420					CKLOOP				
(3)	017420	104406							TRAP	C\$CLP1
2058	017422	005037	003430		4\$:	CLR	TMP2	:CLEAR BAD WORD COUNTER		
2059	017426	012703	004364			MOV	#BUF1,R3	:GOOD DATA STORED IN BUF1		
2060	017432	012702	005364			MOV	#BUF2,R2	:DATA BUFFER WRITTEN INTO BY MAINT.		
2061	017436	005001				CLR	R1	: WORD COUNT.		
2062	017440	011337	003432		5\$:	MOV	(R3),GDDAT	:EXPECTED DATA		
2063	017444	011237	003434			MOV	(R2),BDDAT	:GET DATA FROM BUFFER		
2064	017450	023737	003432	003434		CMP	GDDAT,BDDAT			
2065	017456	001436				BEQ	7\$	:BR IF DATA IS RIGHT.		
2066	017460	010237	003424			MOV	R2,TMP0	: GET ERROR ADDRESS...		
2067	017464	010137	003426			MOV	R1,TMP1	:...AND WORD NUMBER.		
2068	017470	005737	003430			TST	TMP2	: 1ST ERROR ENCOUNTERED ??		
2069	017474	001004				BNE	6\$	:NO, SKIP THE ERROR HEADER.		
2070	017476					DFERR	EM22,ERR0	:ERROR MESSAGE ON 1ST ERROR...		
(5)	017476	104455							TRAP	C\$ERDF
(6)	017500	000224							.WORD	148
(6)	017502	032037							.WORD	EM22
(6)	017504	026164							.WORD	ERR0
2071	017506	005237	003430		6\$:	INC	TMP2	:...FOLLOWED BY FAILING DATA.		
2072	017512					PRINTX	#FRMT14,TMP1,TMP0,GDDAT,BDDAT			
(11)	017512	013746	003434						MOV	BDDAT,-(SP)
(10)	017516	013746	003432						MOV	GDDAT,-(SP)
(9)	017522	013746	003424						MOV	TMP0,-(SP)
(8)	017526	013746	003426						MOV	TMP1,-(SP)
(7)	017532	012746	030031						MOV	#FRMT14,-(SP)

(6)	017536	012746	000005						
(3)	017542	010600						MOV	#5,-(SP)
(4)	017544	104415						MOV	SP,R0
(4)	017546	062706	000014					TRAP	C\$PNTX
2073	017552							ADD	#14,SP
(3)	017552	104406			CKLOOP			TRAP	C\$CLP1
2074	017554	005722		7\$:	TST	(R2)+			
2075	017556	005723			TST	(R3)+			: INCREMENT BUFFER POINTERS.
2076	017560	005201			INC	R1			: AND THE WORD COUNT.
2077	017562	020127	000377		CMP	R1,#255.			: DONE ALL WORDS ??
2078	017566	002724			BLT	5\$			: NOT YET, CONTINUE.
2079	017570	005737	003430		TST	TMP2			: YES, ANY ERRORS LOGGED ??
2080	017574	001412			BEQ	8\$			: NO.
2081	017576				PRINTB	#FRMT15,TMP2			: YES, PRINT SUMMARY.
(8)	017576	013746	003430						
(7)	017602	012746	030120					MOV	TMP2,-(SP)
(6)	017606	012746	000002					MOV	#FRMT15,-(SP)
(3)	017612	010600						MOV	#2,-(SP)
(4)	017614	104414						MOV	SP,R0
(4)	017616	062706	000006					TRAP	C\$PNTB
2082	017622	012737	123456	003432	8\$:	MOV	#123456,GDDAT		
2083	017630	011237	003434			MOV	(R2),BDDAT		: EXPECTED DATA IN LAST WORD+1
2084	017634	023737	003432	003434		CMP	GDDAT,BDDAT		: GET LAST WORD+1 FROM BUF2
2085	017642	001404				BEQ	9\$		
2086	017644					DFERR	EM23,ERR4		: LAST+1 INCORRECT.
(5)	017644	104455							
(6)	017646	000225						TRAP	C\$ERDF
(6)	017650	032103						.WORD	149
(6)	017652	026304						.WORD	EM23
2087	017654				9\$:	ENDSEG		.WORD	ERR4
(3)	017654								
(3)	017654	104405						10000\$:	
2088	017656	000207				RTS	PC		: RETURN.
2089									
2090	017660				10\$:	ENDTST			
(3)	017660								
(3)	017660	104401						L10044:	
								TRAP	C\$ETST

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 PAGE 36
27 -- RLV11/12 MAINTENANCE, FIFO ADDRESSING.

SEQ 0084

```
2092      .SBTTL  27 -- RLV11/12 MAINTENANCE, FIFO ADDRESSING.
2093
2094      017662      STARJ
2095      (2)      ;*****
2096      2095      ;TEST THAT FIFO OPERATES CORRECTLY.  STORE ADDRESS PATTERN
2097      2096      ;IN BUF1 (0-255) THAT CONTAINS A UNIQUE PATTERN IN EACH LOCATION.
2098      2097      ;PERFORM MAINTENANCE FUNCTION AND TEST BUF2 FOR PROPER FIFO
2099      2098      ;ADDRESSING.  NOTE THAT CRC'S ARE NOT CHECKED IN THIS TEST.
2100      2099      ;REPEAT A SECOND TIME USING A COMPLIMENT ADDRESS PATTERN.
2101      2100      ;ALSO USE AND TEST FOR INTERRUPT ON MAINT DONE.
2102      2101      STARS
2103      (2)      ;*****
2104      2102      ;*****
2105      (4)      BEGIN.TEST
2106      2103      T27::
2107      2104      ; SUBTEST 1 -- FIFO ADDRESS PATTERN.
2108      2105      ;
2109      2106      FIFO1:  BGNSUB
2110      (3)      017662
2111      (3)      017662      104402
2112      2107      017664      005001
2113      2108      017666      012702      000400
2114      2109      017672      012703      004364
2115      2110      017676      010123
2116      2111      017700      005201
2117      2112      017702      005302
2118      2113      017704      001374
2119      2114      017706      004737      017754
2120      2115      017712
2121      (3)      017712
2122      (3)      017712      104403
2123      2116      ;
2124      2117      ; SUBTEST 2 -- FIFO COMPLIMENT ADDRESS PATTERN.
2125      2118      ;
2126      2119      FIFO2:  BGNSUB
2127      (3)      017714
2128      (3)      017714
2129      (3)      017714      104402
2130      2120      017716      012701      177777
2131      2121      017722      012702      000400
2132      2122      017726      012703      004364
2133      2123      017732      010123
2134      2124      017734      005301
2135      2125      017736      005302
2136      2126      017740      001374
2137      2127      017742      004737      017754
2138      2128      017746
2139      (3)      017746
2140      (3)      017746      104403
2141      2129      017750
2142      (3)      017750      104432
2143      (3)      017752      000332
2144      2130      ;
2145      2131      ; THIS SUBROUTINE EXECUTES THE MAINT FUNCTION AND TESTS RESULTS.
2146      2132      ;
2147      2133      XFIFO:  BGNSEG
2148      (3)      017754      104404
2149      (3)      017754      104404
```

CLR R1
MOV #256,R2
MOV #BUF1,R3
1\$: MOV R1,(R3)+
INC R1
DEC R2
BNE 1\$
JSR PC,XFIFO
ENDSUB

;SETUP TO STORE PATTERN IN BUF1
;INC. PATTERN
; EXECUTE AND CHECK FIFO DATA.

T27.1: TRAP CSBSUB

L10051: TRAP CSBSUB

T27.2: TRAP CSBSUB

1\$: MOV #-1,R1
MOV #256,R2
MOV #BUF1,R3
1\$: MOV R1,(R3)+
DEC R1
DEC R2
BNE 1\$
JSR PC,XFIFO
ENDSUB

; STORE COMPLIMENT ADDRESS PATTERN.
; EXECUTE AND CHECK FIFO DATA.

L10052: TRAP CSBSUB

EXIT TST ; ALL DONE.
TRAP CSEXIT
.WORD L10050-

*** LOOP SEGMENT ***
TRAP CSBSEG

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 H 7
27 -- RLV11/12 MAINTENANCE, FIFO ADDRESSING.

SEQ 0085

```
2134 017756 012702 000400      MOV    #256,R2      ;SETUP TO CLEAR BUF2
2135 017762 012703 005364      MOV    #BUF2,R3
2136 017766 005023 1$:      CLR    (R3)+
2137 017770 005302      DEC    R2
2138 017772 001375      BNE    1$
2139 017774 005037 010634      CLR    INTFLG      ;CLEAR INT. FLAG
2140 020000      SETPRI  #PRI00
2141 (3) 020000 012700 000000      MOV    #PRI00,R0
2142 (3) 020004 104441      TRAP   C$SPRI
2143 020006 004537 010210      JSR    R5,LDFUN      ;LOAD FUNCTION
2144 020012 000100 177001      MAINT:INTEN,-511.
2145 020016 004537 010640      JSR    R5,WTCRDY     ;WAIT FOR READY
2146 020022 004537 007510      JSR    R5,GETERR     ;CHECK CONTROLLER FOR ERRORS
2147 020030 000405      BR     2$           ;BR IF NONE.
2148 (5) 020030 104455      DFERR   EM98,ERR6    ;UNEXPECTED CONTROLLER ERRORS.
2149 (6) 020032 000226      TRAP   C$ERDF
2150 (6) 020034 034347      .WORD  150
2151 (6) 020036 026340      .WORD  EM98
2152 (3) 020040      .WORD  ERR6
2153 020042 104406      TRAP   C$CLP1
2154 020042 032737 022000 003364 2$: BIT    #OPI!NXM,E.CS  ; OPI OR NXM ??
2155 020050 001110      BNE    8$           ; QUIT IF SO.
2156 020052 005737 010634      TST    INTFLG      ;CHECK FOR INTERRUPT
2157 020056 001005      BNE    4$           ; DIDN'T INTERRUPT.
2158 (5) 020060 104455      DFERR   EM15,ERRO
2159 (6) 020062 000227      TRAP   C$ERDF
2160 (6) 020064 031544      .WORD  151
2161 (6) 020066 026164      .WORD  EM15
2162 (3) 020070      .WORD  ERRO
2163 020070 104406      TRAP   C$CLP1
2164 020072 005037 003430      CKLOOP
2165 020076 012703 004364      4$:  CLR    TMP2      ;CLEAR BAD WORD COUNTER
2166 020102 012702 005364      MOV    #BUF1,R3      ;GOOD DATA STORED IN BUF1
2167 020106 005001      MOV    #BUF2,R2      ;DATA BUFFER WRITTEN INTO BY MAINT.
2168 020110 011337 003432      CLR    R1            ;WORD COUNT.
2169 020114 011237 003434      5$:  MOV    (R3),GDDAT   ;EXPECTED DATA
2170 020120 023737 003432 003434      MOV    (R2),BDDAT   ;GET DATA FROM BUFFER
2171 020126 001436      CMP    GDDAT,BDDAT
2172 020130 010237 003424      BEQ    7$           ;BR IF DATA IS RIGHT.
2173 020134 010137 003426      MOV    R2,TMP0      ;GET ERROR ADDRESS...
2174 020140 005737 003430      MOV    R1,TMP1      ;...AND WORD NUMBER.
2175 020144 001004      TST    TMP2          ;1ST ERROR ENCOUNTERED ??
2176 (5) 020146 104455      BNE    6$           ;NO, SKIP THE ERROR HEADER.
2177 (6) 020150 000230      DFERR   EM22,ERRO    ;ERROR MESSAGE ON 1ST ERROR...
2178 (6) 020152 032037      TRAP   C$ERDF
2179 (6) 020154 026164      .WORD  152
2180 020156 005237 003430      .WORD  EM22
2181 (11) 020162 013746 003434      .WORD  ERRO
2182 (10) 020166 013746 003432      6$:  INC    TMP2      ;...FOLLOWED BY FAILING DATA.
2183 (9) 020172 013746 003424      PRINTX #FRMT14,TMP1,TMP0,GDDAT,BDDAT
2184 (8) 020176 013746 003426      MOV    BDDAT,-(SP)
2185 (7) 020202 012746 030031      MOV    GDDAT,-(SP)
2186      MOV    TMP0,-(SP)
2187      MOV    TMP1,-(SP)
2188      MOV    #FRMT14,-(SP)
```

SEQ 0086

Address	Hex	Dec	Label	Instruction	Comment	Register	Value
2169	020206	012746	000005			MOV	#5,-(SP)
	(3)	020212	010600			MOV	SP,R0
	(4)	020214	104415			TRAP	C\$PNTX
	(4)	020216	062706	000014		ADD	#14,SP
2170	020222	104406		CKLOOP		TRAP	C\$CLP1
2171	020224	005722	7\$:	TST (R2)+	; INCREMENT BUFFER POINTERS.		
2172	020226	005723		TST (R3)+			
2173	020230	005201		INC R1	; AND THE WORD COUNT.		
2174	020232	020127	000377	CMP R1,#255.	; DONE ALL WORDS ??		
2175	020236	002724		BLT 5\$	; NOT YET, CONTINUE.		
2176	020240	005737	003430	TST TMP2	; YES, ANY ERRORS LOGGED ??		
2177	020244	001412		BEQ 8\$	; NO.		
	(8)	020246	013746	PRINTB #FRMT15,TMP2	; YES, PRINT SUMMARY.		
	(7)	020252	012746			MOV	TMP2,-(SP)
	(6)	020256	012746			MOV	#FRMT15,-(SP)
	(3)	020262	010600			MOV	#2,-(SP)
	(4)	020264	104414			MOV	SP,R0
	(4)	020266	062706	000006		TRAP	C\$PNTB
2178	020272		8\$:	ENDSEG		ADD	#6,SP
	(3)	020272					
	(3)	020272	104405			10000\$:	
2179	020274			SETPRI #PRI07		TRAP	C\$ESEG
	(3)	020274	012700				
	(3)	020300	104441			MOV	#PRI07,R0
2180	020302	000207		RTS PC	; RETURN.	TRAP	C\$SPRI
2181	020304			ENDTST			
	(3)	020304				L10050:	
	(3)	020304	104401			TRAP	C\$ETST

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 J 7 PAGE 37
28 -- RLV11/12 MAINTENANCE, BANK 7 SELECT AND NEXM TEST.

SEQ 0087

.SBTTL 28 -- RLV11/12 MAINTENANCE, BANK 7 SELECT AND NEXM TEST.

STARS

* FOR RLV12:
* TEST THAT BBS7 WILL SELECT THE I/O PAGE AND THAT ACCESS TO
* LOCATION 0 IN THAT PAGE WILL GENERATE NXM AND OPI ERRORS.
* NOTE -- IF BANK 7 IS NOT PROPERLY SELECTED, 1000 BYTES
* STARTING AT XXXX1000 WILL PROBABLY GET CPUNCHED !!!!

* FOR RLV11:
* RLV11 DOESN'T ASSERT BBS7, SO WE'LL THIS TEST
* IF PHYSICAL MEMORY SIZE IS 124K OR GREATER.

STARS

BEGIN.TEST

T28::

CMP RLTP,#RLV11
BGT 1\$
CMP MSIZE,#124.
BLT 2\$
EXIT TST

; EXECUTE ALWAYS ON RLV12...
;...AND ON RLV11 IF < 124KW.

TRAP .WORD C\$EXIT
L10053-

1\$: MOVB #',,EMBOX

; ADJUST ERROR TEXT FOR RLV12...

2\$: SKP2

...OR FOR RLV11.
DO CRC FOR INITIAL...

CLRB

...DA OF 0.

JSR

POINT TO LOC 0 IN LAST 4K.
SET AN 18 BIT...

R5,CALCRC

0

MOV

#160000,BA16

MOV

#3,R0

CMP

ABUSW,#22.

BNE

3\$

MOV

#77,R0

MOV

R0,BA22

JSR

R5,LDIUN

MAINT,-511.

JSR

R5,WTCDY

MOV

#ERR!NXM,R0

BIC

E.CS,R0

BEG

4\$

...OR 22 BIT EXTENSION.

XCT MAINTENANCE...

...WITH VALID WORD COUNT.

EXPECT AT LEAST THESE ERRORS.

EXIT IF WE'RE OK.

OTHERWISE, GET WHAT'S THERE.

AND GET PHYSICAL ADDRESS.

BBS7 AND/OR NXM FAILURE.

TRAP .WORD C\$ERDF
153
EM80
ERR12

4\$: MOV #BUF1,BA16
CLR BA22
ENDTST

; RESET BA16 AND 22.

L10053:

TRAP C\$ETST

2184
2185
2186 020306
(2)

2187
2188
2189
2190
2191
2192
2193
2194
2195 020306
(2)

2196 020306
(4)

2197 020306 023727 003402 000001

2198 020306 003006

2199 020314 023727 011232 000174

2200 020316 007406

2201 020324 104432

2202 020326 000150

2203 020326 112757

2204 020326 000402

2205 020326 105037

2206 020326 004537

2207 020326 000000

2208 020326 012757

2209 020326 160000 010352

2210 020326 000003

2211 020326 023727 011240 000026

2212 020326 001002

2213 020326 012700 000077

2214 020402 010037 010354

2215 020402 004537 010210

2216 020412 000000 177001

2217 020412 004537 010640

2218 020422 012700 120000

2219 020422 043700 003364

2220 020432 001415

2221 020434 004537 007510

2222 020440 000240

2223 020442 013737 010354 003426

2224 020450 013737 010352 003430

2225 020456 104455

(5) 020456 000231

(6) 020460 034061

(6) 020462 026500

2226 020466 012737 004364 010352 4\$:

2227 020474 005037 010354

2228 020500

(3) 020500

(3) 020500 104401

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 K 7 PAGE 38
29 -- RLV11/12 MAINTENANCE, EXTENDED MEMORY ACCESS TEST.

SEQ 0088

```
2230 .SBTTL 29 -- RLV11/12 MAINTENANCE, EXTENDED MEMORY ACCESS TEST.
2231
2232 020502 STARS
2233 *****
2234 * BYPASS THIS TEST IF NO KT SUPPORT.
2235 * OTHERWISE, MEMORY HAS ALREADY BEEN SIZED.
2236 * PAR'S 0 TO 6 ARE MAPPED TO THE 1ST 28K, AND PAR7 TO THE I/O PAGE.
2237 *
2238 * NOW ACCESS EXTENDED MEMORY IN 4K INCREMENTS. STORE A RANDOM DATA
2239 * PATTERN IN BUF1 (RELOCATED). EXECUTE MAINTENANCE MODE, AND CHECK
2240 * BUF2 (RELOCATED) FOR CORRECT DATA TRANSFER. ALSO CHECK THAT THE FINAL
2241 * BUS ADDRESS INCREMENTED CORRECTLY (OVERFLOWING INTO THE BAE).
2242 * WHEN THE END OF PHYSICAL MEMORY IS REACHED, A BUS-ERROR WILL OCCUR
2243 * DURING THE BUFFER SET-UP PHASE AND WE'LL LOOK FOR A NXM ERROR AT
2244 * THE SAME PHYSICAL ADDRESS. CONTINUE UNTIL WE REACH THE I/O PAGE
2245 * AT 760000 (18 BIT), OR 17760000 (22 BIT).
2246 STARS
2247 *****
2248 PAR4= KIPAR0+10 ; USE KIPAR4 FOR ACCESSING.
2249 BEGIN.TEST
2250
2251 T29::
2252 CLR TRPFLG ; CLEAR TRAP FLAG.
2253 TST MMRO ; KT AVAILABLE ??
2254 240
2255 TST TRPFLG
2256 BEQ EXMTST ; PROCEED IF SO.
2257 1$: EXIT TST ; ...EXIT OTHERWISE.
2258
2259 TRAP WORD C$EXIT
2260 EXMTST: MOV #7400, IOPAG ; ASSUME 18 BITS, SET LIMIT AT 124K...
2261 CMP ABUSW, #22. ; 22 BITS AVAILABLE ??
2262 BNE 1$ ; BR IF NOT.
2263 CMP RLTP, #RLV12X ; CAN WE HANDLE THEM ALL ??
2264 BNE 1$ ; BR IF NOT.
2265 MOV #20, MMRO ; YES, SET KT IN 22 BIT MODE...
2266 MOV #177400, IOPAG ; ...AND RAISE LIMIT TO 2044K.
2267 1$: CLR BA22
2268 MOV #177000, BA16 ; SET PHYSICAL ADDRESS AT 32K-256 WORDS.
2269 MOV #1770, PAR4 ; SET PAR4 TO THE SAME POINT.
2270 EXMLUP: JSR R5, SETRAN ; RANDOM DATA TO BUF1, ZERO BUF2.
2271 BGNSEG
2272
2273 TRAP C$BSL
2274 MOV #BUF1, R1 ; SETUP TO COPY FROM BUF1...
2275 MOV #100000, R2 ; ...TO (PAR4)+0
2276 MOV #512, R3
2277 MOV #228, ERRVEC ; SOONER OR LATER WE'LL TRAP.
2278 CLR TRPFLG
2279 INC MMRO
2280 2$: MOV (R1)+, (R2)+ ; ***** KT ON *****
2281 240, 240 ; COPY BUFFERS VIA PAR4.
2282 SOB R3, 2$ ; LOOP 'TIL DONE.
2283 BR 3$
2284 22$: MOV R2, TRPFLG ; NXM TRAP, SAVE ADDRESS...
2285 MOV #3$, (SP)
2286 RTI ; ...AND CONTINUE.
2287 3$: CLR MMRO ; ***** KT OFF *****
```


CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 L 7
29 -- RLV11/12 MAINTENANCE, EXTENDED MEMORY ACCESS TEST.

SEQ 0089

```
2280 020700 012737 010624 000004 MOV #TRAP4,ERRVEC ; RESET TRAP.
2281 020706 004537 010714 JSR R5,CALCRC ; JUST TO BE CONSISTENT.
2282 020712 000000 0 JSR R5,LDFUN
2283 020714 004537 010210 MAINT, -511 ; EXECUTE MAINT MODE DMA TO EXT MEM.
2284 020720 000000 177001 JSR R5,WTCRDY
2285 020724 004537 010640 CMP R1,TYP,#RLV12X ; BAE IN USE ??
2286 020730 023727 004402 000003 BEQ 33$ ; BR IF SO.
2287 020736 001411 003364 MOV E,CS,R0 ; NO, GET EA<17:16> FROM FINAL CSR.
2288 020740 013700 003364 ASR R0
2289 020744 006200 ASR R0
2290 020746 006200 ASR R0
2291 020750 006200 ASR R0
2292 020752 006200 BIC #^C3,R0
2293 020754 042700 177774 SKP2
2294 020760 000402 003376 33$: MOV E,BAE,R0 ; GET EA<21:16> FROM BAE.
2295 020762 013700 003426 MOV R0,TMP1 ; SAVE FINAL PA IN CASE...
2296 020766 010037 003366 003430 MOV E,BA,TMP2 ; ...WE HAVE A STATUS ERROR.
2297 020772 013737 010626 TST TRPFLG ; NOW, ARE WE EXPECTING NXM ??
2298 021000 005737 010626 BEQ 4$ ; BR IF NOT.
2299 021004 001415 120000 MOV #ERR!NXM,R0 ; YES, DO WE HAVE IT ??
2300 021006 012700 003364 BIC E,CS,R0
2301 021012 043700 007510 BEQ 8$ ; BR IF SO.
2302 021016 001501 007510 JSR R5,GETERR ; NO, GET WHATEVER'S THERE.
2303 021020 004537 000240 NOP
2304 021024 000240 DFERR EM90,ERR12 ; EXTENDED NXM STATUS WRONG.
2305 021026 104455 (5) 021026 000232 TRAP CSERDF
2306 021030 034147 (6) 021030 026500 .WORD 154
2307 021032 000471 (6) 021032 000471 .WORD EM90
2308 021034 000471 (6) 021034 000471 .WORD ERR12
2309 021040 004537 007510 4$: JSR R5,GETERR ; ANY OTHER STATUS ERRORS ??
2310 021044 000405 007510 BR 44$ ; BR IF NOT.
2311 021046 104455 DFERR EM91,ERR13 ; STATUS INCORRECT ON EXT MEMORY ACCESS.
2312 021050 006233 (5) 021050 034223 TRAP CSERDF
2313 021052 034223 (6) 021052 026534 .WORD 155
2314 021054 000461 (6) 021054 000461 .WORD EM91
2315 021056 000461 (6) 021056 000461 .WORD ERR13
2316 021060 012701 100000 44$: MOV #100000,R1 ; POINT TO BUF1 IN EXT MEMORY.
2317 021064 012702 101000 MOV #101000,R2 ; DITTO BUF2.
2318 021070 005003 CLR R3 ; WORD NUMBER.
2319 021072 012704 000377 MOV #255.,R4 ; LOOP COUNT.
2320 021076 005237 177572 INC MMRO ***** KT ON *****
2321 021102 012137 003432 5$: MOV (R1)+,GDDAT ;
2322 021106 012237 003434 MOV (R2)+,BDDAT ;
2323 021112 023737 003432 003434 CMP GDDAT,BDDAT ; COMPARE DATA.
2324 021120 001002 BNE 6$ ; BR IF WRONG.
2325 021122 005203 INC R3 ; BUMP WORD COUNT.
2326 021124 077412 SOB R4,5$ ; LOOP.
2327 021126 005037 177572 6$: CLR MMRO ***** KT OFF *****
2328 021132 005704 TST R4 ; LOOP COMPLETED (NO ERRORS) ??
2329 021134 001432 BEQ 8$ ; BR IF SO.
2330 021136 010337 003424 MOV R3,TMP0 ; NO, SAVE FAILING WORD NUMBER.
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 M 7
29 -- RLV11/12 MAINTENANCE, EXTENDED MEMORY ACCESS TEST.

SEQ 0090

2328	021142	012703	000006		MOV	#6,R3			
2329	021146	013737	172350	003430	MOV	PAR4,TMP2			
2330	021152	005037	003426		CLR	TMP1			
2331	021160	006337	003430		ASL	TMP2	:	NOW ASSEMBLE BAD PA,	
2332	021164	006137	003426	7\$:	ROL	TMP1	:	ADJUST PAGE TO BIT 21...	
2333	021170	077305			S08	R3,7\$			
2334	021172	042702	160000		BIC	#(17777,R2	:	...STRIP PAR BITS FROM VA...	
2335	021176	162702	000002		SUB	#2,R2			
2336	021202	060237	003430		ADD	R2,TMP2	:	...AND COMBINE THE PIECES.	
2337	021206	005537	003426		ADC	TMP1			
2338	021212				DFERR	EM92,ERR14	:	DATA ERROR IN EXTENDED MEM.	
(5)	021212	104455						TRAP	C\$ERDF
(6)	021214	000234						.WORD	156
(7)	021216	034300						.WORD	EM92
(8)	021220	026544						.WORD	ERR14
2339									
2340	021222			8\$:	CKLOOP			TRAP	C\$CLP1
(3)	021222	104406			ENDSEG				
2341	021224							10000\$:	
(3)	021224							TRAP	C\$ESEG
(3)	021224	104405							
2342	021226	062737	020000	010352	ADD	#20000,BA16	:	INCR PHYSICAL ADDRESS BY 4K...	
2343	021234	005537	010354		ADC	BA22	:	...AND CVERFLOW INTO BAE.	
2344	021240	062737	000200	172350	ADD	#200,PAR4	:	INCR PAR4 BY 4K ALSO.	
2345	021246	023727	172350	9\$:	CMP	PAR4,(PC)+	:	QUIT AT THE I/O PAGE...	
2346	021252	007400		10PAG:	7400		:	...007400 (124k)...	
2347							:	...OR 177400 (2044k).	
2348	021254	103002			BHIS	EXMDUN	:	EXIT IF DONE.	
2349	021256	000137	020610		JMP	EXMLUP	:	LOOP OTHERWISE.	
2350									
2351	021262	012737	004364	010352	EXMUN:	MO	:	RESET BA16 AND 22.	
2352	021270	005037	010354		CLR	#BUF1,BA16			
2353	021274				ENDTS1	BA22			
(3)	021274							L10054:	
(3)	021274	104401						TRAP	C\$ETST

```

2355      .SBTTL
2356      .SBTTL RLV12 DRIVE INTERFACE TESTS (G5388 TLM REQUIRED).
2357      .SBTTL
2358 021276 STARS
2359      *****
2360      * ALL THESE TESTS REQUIRE THE SERVICES OF THE G5388 TEST LOOP MODULE.
2361      .SBTTL 30 -- SYS CLK, PWR OK, DRIVE SELECT, READY, AND ERROR BITS.
2362      *
2363      * 1. TEST SYS CLOCK AND PWR OK FROM THE RLV12.
2364      * 2. TEST DRIVE READY AND DRIVE ERROR TO THE RLV12.
2365      * 3. TEST DRIVE SELECT BITS FROM THE RLV12.
2366 021276 STARS
2367      *****
2368      TLM1: BEGIN.TEST
2369      (4) 021276      004737 023762      JSR      PC,TLMOK      : CHECK FOR V12 AND TLM OK...
2370      (4) 021276      102561      BVS      7$          : ...AND EXIT IF NOT.
2371      (4) 021276      012777 000001 162032  MOV      #1,@TCSR      : RESET...
2372      (4) 021276      005077      CLR      @TCSR        : ...AND CLEAR TLM CSR (LD BYTE).
2373      (4) 021276      017737 162022 003434  MOV      @TCSR,BDDAT    : GET TLM STATUS.
2374      (4) 021276      012737 004400 003432  MOV      #4400,GDDAT    : EXPECT SYSCLK<11> AND PWROK<8>.
2375      (4) 021276      023737 003432 003434  CMP      GDDAT,BDDAT
2376      (4) 021276      001404      BEQ      1$          : BR IF STATUS IS RIGHT.
2377      (4) 021276      104455      DFERR    EM101,ERR2      : PWR-OK AND/OR SYS-CLK NOT SET IN TLM.
2378      (4) 021276      000237      TRAP     157          : TRAP 157
2379      (4) 021276      035070      .WORD    EM101          : .WORD EM101
2380      (4) 021276      026220      .WORD    ERR2          : .WORD ERR2
2381      (4) 021276      017700      161746      1$:  MOV      @RLCS,RO      : GET RLV12 CSR.
2382      (4) 021276      003434      MOV      RO,BDDAT      : REC'D RLV12 STATUS...
2383      (4) 021276      010037 000001      BIC      #DRDY,RO      : ...SHOULD HAVE DRDY CLEAR.
2384      (4) 021276      010037 003432 003434  MOV      RO,GDDAT
2385      (4) 021276      023737 003432 003434  CMP      GDDAT,BDDAT
2386      (4) 021276      001404      BEQ      2$          : BR IF SO.
2387      (4) 021276      104455      DFERR    EM102,ERR2      : DRIVE READY NOT CLEAR IN RLV12.
2388      (4) 021276      000237      TRAP     158          : TRAP 158
2389      (4) 021276      035070      .WORD    EM102          : .WORD EM102
2390      (4) 021276      026220      .WORD    ERR2          : .WORD ERR2
2391      (4) 021276      017700      000200 161724 2$:  MOV      #200,@TCSR      : SET DRIVE READY IN TLM.
2392      (4) 021276      017700 161700      MOV      @RLCS,RO      : GET RLV12 AGAIN.
2393      (4) 021276      010037 003434      MOV      RO,BDDAT
2394      (4) 021276      035070 000001      BIS      #DRDY,RO      : DRDY SHOULD BE SET NOW.
2395      (4) 021276      010037 003432 003434  MOV      RO,GDDAT
2396      (4) 021276      023737 003432 003434  CMP      GDDAT,BDDAT
2397      (4) 021276      001404      BEQ      3$          : BR IF SO.
2398      (4) 021276      104455      DFERR    EM102,ERR2      : DRIVE READY NOT SET IN RLV12.
2399      (4) 021276      000237      TRAP     159          : TRAP 159
2400      (4) 021276      035070      .WORD    EM102          : .WORD EM102
2401      (4) 021276      026220      .WORD    ERR2          : .WORD ERR2
2402      (4) 021276      017700      000100 161656 3$:  MOV      #100,@TCSR      : CLEAR DRDY, SET DERR IN TLM.
2403      (4) 021276      017700 161632      MOV      @RLCS,RO      : ONE MORE TIME.
2404      (4) 021276      010037 003434      MOV      RO,BDDAT
2405      (4) 021276      052700 140000      BIS      #ERR:DERR,RO  : SHOULD HAVE THESE ERROR BITS.

```

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 ⁸ PAGE 39-1
30 -- SYS CLK, PWR OK, DRIVE SELECT, READY, AND ERROR BITS.

SEQ 0092

```
2396 021502 010037 003432      MOV    R0,GDDAT
2397 021506 023737 003432 003434  CMP    GDDAT,BDDAT
2398 021514 001404      BEQ    48
2399 021516      DFERR    EM103,ERR2      ; BR IF SO.
                                     ; DRIVE ERROR STATUS INCORRECT IN RLV12.
(5) 021516 104455      TRAP    C$ERDF
(6) 021520 000240      .WORD 160
(6) 021522 035145      .WORD EM103
(6) 021524 026220      .WORD ERR2
2400 021526 005077 161612 48:  CLR    @TCSR      ; CLEAR TLM.
2401 021532 005001      CLR    R1        ; INIT DRIVE SELECT BITS<9:8>...
2402 021534 112737 000060 035235  MOVB   #'0,EM104X  ; ...AND ERROR TEXT.
2403 021542 012703 000004      MOV    #4,R3      ; LOOP CONTROL.
2404 021546 012777 000200 161550 58:  MOV    #CRDY,@RLCS  ; INIT RLV CSR...
2405 021554 050177 161544      BIS    R1,@RLCS   ; ...INSERT DRIVE BITS <9:8>.
2406 021560 017702 161560      MOV    @TCSR,R2   ; GET TLM CSR.
2407 021564 010237 003434      MOV    R2,BDDAT   ; RECV'D TLM SHOULD ECHO DRIVE...
2408 021570 042702 003000      BIC    #BIT10!BIT9,R2 ; ...SELECTED IN TLM<10:9>.
2409 021574 006301      ASL    R1
2410 021576 050102      BIS    R1,R2      ; SET EXPECTED VALUE.
2411 021600 006201      ASR    R1
2412 021602 010237 003432      MOV    R2,GDDAT
2413 021606 023737 003432 003434  CMP    GDDAT,BDDAT
2414 021614 001404      BEQ    68
2415 021616      DFERR    EM104,ERR2      ; DRIVE SELECT FAILS IN TLM.
(5) 021616 104455      TRAP    C$ERDF
(6) 021620 000241      .WORD 161
(6) 021622 035227      .WORD EM104
(6) 021624 026220      .WORD ERR2
2416 021626 062701 000400 68:  ADD     #DS1,R1      ; BUMP DRIVE SELECT NUMBER...
2417 021632 105237 035235      INCB   EM104X    ; ...AND THE ERROR TEXT.
2418 021636 077335      SOB    R3,58
2419 021640 012777 000200 161456 78:  MOV    #CRDY,@RLCS  ; RESET DRIVE NUMBER IN RL.
2420 021646      ENDTST
(5) 021646      L10055:
(5) 021646 104401      TRAP    C$ETST
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 C 8
31 -- DRIVE COMMAND, STATUS AND STATUS CLOCK.

SEQ 0093

2422
2423
2424 021650

(2)
2425
2426
2427 021650

(2)
2428 021650

(4)
2429 021650
2430 021654

004737 023762
2431 021656 004737 024132
2432 021662 005037 003436

012765 125252 000004
2433 021666 012765 161422
2434 021674 012777 000204 161422

012777 000013 161420
2435 021702 012777 010016
2436 021710 004537 000200 161402

042777
2437 021714 042777
2438 021722 005000

011501
2439 021724 011501
2440 021726 100401

077003
2441 021730 077003
2442 021732 016537 000002 003434

010640
2443 021740 004537 010640
2444 021744 012737 000013 003432

003432 003434
2445 021752 023737 003432 003434
2446 021760 001404

021762
2447 021762 104455
(5) 021762 104455

000242
(6) 021764 000242
(6) 021766 035311

026304
(6) 021770 026304
2448 021772 013737 003372 003434

012737 125252 003432
2449 022000 012737 003432 003434
2450 022006 023737 003432 003434

001404
2451 022014 001404
2452 022016 104455

000243
(5) 022016 104455
(6) 022020 000243

035373
(6) 022022 035373
(6) 022024 026304

004537 007510
2453 022026 004537 007510
2454 022032 000407

012737 036670 003426
2455 022034 012737 036670 003426
2456 022042 104455

(5) 022042 104455
(6) 022044 000244
(6) 022046 035443

026666
(6) 022050 026666
2457 022052 104401
(3) 022052 104401
(3) 022052 104401

.SBTTL 31 -- DRIVE COMMAND, STATUS AND STATUS CLOCK.

STARS

: * NOW TEST DRIVE COMMAND, STATUS, AND STATUS CLOCK
: * USING A 'GET STATUS' COMMAND.
STARS

TLM2: BEGIN.TEST

T31::

JSR PC,TLMOK
BVS 5\$: EXIT IF NO GOOD.
JSR PC,CLRTLM : RESET TLM, AND SET DRDY.
CLR INIMP
MOV #125252,4(R5) : SET PHONEY STATUS WORD.
MOV #CRDY!GSTAT,RLCS : SET 'GET STATUS' COMMAND...
MOV #DRST!GSBIT!MK,RLDA : ...AND RESET STATUS BITS.
JSR R5,BEFORE
BIC #CRDY,RLCS : XCT IT.
CLR R0
1\$: MOV (R5),R1 : GET TCSR...
BMI 2\$: ...PROCEED WHEN 'NEW SKGS' SETS...
SOB R0,1\$: ...BUT DON'T WAIT FOREVER !!!
2\$: MOV 2(R5),BDDAT : SAVE RECV'D DRIVE COMMAND (SKGS)...
JSR R5,WT(CRDY
MOV #DRST!GSBIT!MK,GDDAT : ...SHOULD LOOK LIKE THIS.
CMP GDDAT,BDDAT
BEQ 3\$
DFERR EM105,ERR4 : BR IF STATUS COMMAND WAS RIGHT.
: GET STATUS COMMAND WRONG IN TLM.

TRAP C\$ERDF
.WORD 162
.WORD EM105
.WORD ERR4

3\$: MOV E.MP,BDDAT : RETURNED DRIVE STATUS...
MOV #125252,GDDAT : ...SHOULD = PHONEY.
CMP GDDAT,BDDAT
BEQ 4\$
DFERR EM106,ERR4 : BR IF RETURNED STATUS IS RIGHT.
: RETURNED STATUS INCORRECT IN RLMP.

TRAP C\$ERDF
.WORD 163
.WORD EM106
.WORD ERR4

4\$: JSR R5,GETERR : FINAL CONTROLLER STATUS...
BR 5\$: ...SHOULD BE ERROR FREE.

MOV #EXPNON,TMP1
DFERR EM107,ERR20A : RLV12 ERRORS AFTER RESET GET STATUS

TRAP C\$ERDF
.WORD 164
.WORD EM107
.WORD ERR20A

5\$: ENDTST

L10056:

TRAP C\$ETST

MACY11 30G(1063) 08-FEB-82 10:07 ^{D 8} PAGE 41
32 -- DRIVE COMMAND, SEEK DIFF AND SECTOR PULSE.

2459				
2460				
2461	022054			
(2)				
2462				
2463	022054			
(2)				
2464	022054			
(4)	022054			
2465	022054	004737	023762	
2466	022060	102455		
2467	022062	004737	024132	
2468	022066	005037	003436	
2469	022072	052715	000024	
2470	022076	012777	000206	161220
2471	022104	012777	077601	161216
2472	022112	004537	010016	
2473	022116	042777	000200	161200
2474	022124	005000		
2475	022126	011501		
2476	022130	100401		
2477	022132	077003		
2478	022134	016537	000002	003434
2479	022142	004537	010640	
2480	022146	012737	077601	003432
2481	022154	023737	003432	003434
2482	022162	001404		
2483	022164			
(5)	022164	104455		
(6)	022166	000245		
(6)	022170	035516		
(6)	022172	026304		
2484	022174	004537	007510	
2485	022200	000407		
2486	022202	012737	036670	003426
2487	022210			
(5)	022210	104455		
(6)	022212	000246		
(6)	022214	035577		
(6)	022216	026666		
2488	022220			
(3)	022220			
(3)	022220	104401		

.SBTTL 32 -- DRIVE COMMAND, SEEK DIFF AND SECTOR PULSE.

STARS

```
*****
* TEST SECTOR PULSE USING A "SEEK" COMMAND.
```

STARS

```
*****
fLM3:  BEGIN.TEST
```

FLM3: BEGIN TEST

132::

```

132::
JSR    PC,TLMOK
BVS    4$                                : BR IF NOT OK.
JSR    PC,CLR TLM                        : RESET TLM, SET DRDY.
CLR     IN|MP
BIS     #24,(R5)                        : ENABLE SECTOR GENERATOR (NOM CLOCK).
MOV     #CRDY!SEEK,@RLCS                : SET "SEEK" COMMAND...
MOV     #77600!MK,@RLDA                : ...AND A PHONEY SEEK DIFFERENCE.
JSR     R5,BEFORE
BIC     #CRDY,@RLCS                    : XCT IT.
CLR     R0
1$:    MOV     (R5),R1
BMI     2$                                : PROCEED WHEN 'NEW SKGS' SETS...
SOB     R0,1$                          : ...KEEP-ALIVE !!
2$:    MOV     2(R5),BDDAT                : SAVE RECV'D DRIVE COMMAND...
JSR     R5,WT(CRDY
MOV     #77600!MK,GDDAT                : ...SHOULD = PHONEY DIFFERENCE.
CMP     GDDAT,BDDAT
BEQ     3$                                : BR IF SO.
DFERR   EM:10,ERR4                      : DRIVE COMMAND WRONG IN TLM.

```

18:

28:

38:

48:

```
TRAP      CSERDF
.WORD     165
.WORD     EM110
.WORD     ERR4
```

```
TRAP      CSERDF
.WORD     166
.WORD     EM111
.WORD     ERR20A
```

L10057:

TRAP CSETST

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 E 8
33 -- WRITE GATE, WRITE GATE ERROR, AND WRITE DATA.

SEQ 0095

2490
2491
2492 022222
(2)
2493
2494 022222
(2)
2495 022222
(2)
2496 022222
(2)
2497 022222
(2)
2498 022222
(2)
2499 022222
(2)
2500 022222
(2)
2501 022222
(2)
2502 022222
(2)
2503 022222
(2)
2504 022222
(2)
2505 022222
(2)
2506 022222
(2)
2507 022222
(2)
2508 022222
(2)
2509 022222
(2)
2510 022222
(2)
2511 022222
(2)
2512 022222
(2)
2513 022222
(2)
2514 022222
(2)
2515 022222
(2)
2516 022222
(2)
2517 022222
(2)
2518 022222
(2)
2519 022222
(2)
2520 022222
(2)
2521 022222
(2)
2522 022222
(2)
2523 022222
(2)
2524 022222
(2)
2525 022222
(2)
2526 022222
(2)
2527 022222
(2)
2528 022222
(2)
2529 022222
(2)
2530 022222
(2)
2531 022222
(2)
2532 022222
(2)
2533 022222
(2)
2534 022222
(2)
2535 022222
(2)
2536 022222
(2)
2537 022222
(2)
2538 022222
(2)
2539 022222
(2)
2540 022222
(2)

004737 023762
102524
004737 024132
052715 000024
004537 010160
000002
012777 000212 161050
012777 177600 003436
013777 003436 161042
012777 004364 161030
012777 000205 161024
004537 010016
042777 000200 161006
005000
011501
010137 003434
052701 030000
010137 003432
023737 003432 003434
001410
077014
004537 010640
104455
000247
035644
026304
000402
004537 010640
011537 003434
013737 003434 003432
042737 060000 003432
023737 003432 003434
001415
032737 020000 003434
001405
104455
000250
036010
026304
000404
104455
000251
035736
026304

.SBTTL 33 -- WRITE GATE, WRITE GATE ERROR, AND WRITE DATA.

STARS

* TEST THE WRITE GATE, WRITE DATA ACTIVE, AND WRITE GATE ERROR.

STARS

* LM4: BEGIN TEST

T33::

JSR PC,TLMOK
BVS 58 ; BR IF NOT OK.
JSR PC,CLRTLM ; RESET TLM, SET DRDY.
BIS #24,(R5) ; ENABLE SECTOR GEN (NOM CLOCK).
JSR R5,WDELAY ; DELAY SO THAT WE ENTER THE TLM...
2 ; ...SEQUENCE ASYNCHRONOUSLY.
MOV #CRDY!WRITE, @RLCS ; SET COMMAND TO WRITE...
MOV #-128, INIMP
MOV INIMP, @RLMP ; ...128 WORDS...
MOV #BUF1, @RLBA ; ...FROM BUF1...
MOV #205, @RLDA ; ...TO CYL 1, SECT 5.
JSR R5,BEFORE
BIC #CRDY, @RLCS ; XCT IT.
C'R R0
18: MOV (R5), R1 ; TLM STATUS => R1.
MOV R1, BDDAT ; RECV'D TLM STATUS...
BIC #BIT13!BIT12, P1 ; ...WILL GET WRITE GATE AND DATA ACTIVE.
MOV R1, GDDAT
CMP GDDAT, BDDAT
BEQ 28 ; BR IF AND WHEN THAT OCCURS...
SOB R0, 18 ; ...WITHIN REASON -- OF COURSE.
JSR R5, WTCRDY
DFERR EM112, ERR4 ; WRITE GATE AND/OR DATA ACTIVE NOT SET.
TRAP CSERDF
.WORD 167
.WORD EM112
.WORD ERR4
SKP2
28: JSR R5, WTCRDY
MOV (R5), BDDAT ; FINAL TLM STATUS...
MOV BDDAT, GDDAT
BIC #BIT14!BIT13, GDDAT ; ...ERROR AND GATE SHOULD BE CLEAR.
CMP GDDAT, BDDAT
BEQ 48 ; BR IF SO.
BIT #BIT13, BDDAT ; IT'S NOT, IS GATE STILL SET ??
BEQ 38
DFERR EM113A, ERR4 ; WRITE GATE STILL SET AFTER DONE.
TRAP CSERDF
.WORD 168
.WORD EM113A
.WORD ERR4
BR 48
38: DFERR EM113, ERR4 ; WRITE GATE ERROR IN TLM.
TRAP CSERDF
.WORD 169
.WORD EM113
.WORD ERR4

CVRLBAO -- RLV12 DISKLESS				MACY11 306(1063) 08-FEB-82 10:07 F 8		PAGE 42-1		
CVRLBA.P11 08-FEB-82 09:59				33 -- WRITE GATE, WRITE GATE ERROR, AND WRITE DATA.				SEQ 0096
2531	022454	004537	007510	48:	JSR	R5,GETERR	: FINAL RLV STATUS...	
2532	022460	000407			BR	SS	:...SHOULD BE ERROR FREE.	
2533	022462	012737	036670 003426		MOV	#EXPNO, TMP1		
2534	022470				DFERR	EM114,ERR20A	: RL ERROR AFTER WRITE.	
(5)	022470	104455						TRAP CSEDF
(6)	022472	000252						.WORD 170
(6)	022472	036066						.WORD EM114
(6)	022476	026666						.WORD ERR20A
2535	022500			58:	ENDTST			
(3)	022500							L10060:
(3)	022500	104401						TRAP CSETST

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 PAGE 43
34 -- READ DATA, READ HEADER, AND READ DATA W/O HEADER.

SEQ 0097

```
2537 .SBTTL 34 -- READ DATA, READ HEADER, AND READ DATA W/O HEADER.
2538
2539 022502 STARS
2540 *****
2541 * READ AND VERIFY DATA ON EACH OF 6 TLM SECTORS.
2542 * SECTOR 3 GETS A READ HEADER AND READ DATA W/O HEADER SEQUENCE.
2543 * REPEAT USING ALTERNATE VCO SETTINGS (WITH MAX AND MIN PEAK SHIFT).
2544 * VARIOUS PEAK SHIFT, CLOCK, AND SECTOR OPTIONS MAY BE SELECTED VIA
2545 * THE "CHANGE SOFTWARE" DIALOGUE AT START/RESTART TIME.
2546 STARS
2547 *****
2548 TLM5: BEGIN.TEST
2549
2550 T34::
2551 JSR PC,TLMPID ; VERIFY TLM OK AND PROM ID.
2552 BVS 8$ ; EXIT IF NOT OK.
2553 MOV #-128,INIMP ; INIT WORD COUNT.
2554 MOV SNGLSEC,12$ ; GET SECTOR NUMBER (IF ANY).
2555 BIC #^C7,12$
2556 BNE 11$ ; NZ = SINGLE SECTOR MODE.
2557 INC 12$ ; OTHERWISE, USE ALL STARTING WITH 1.
2558 JSR PC,PSX ; SET BUFFERS AND DISK ADDRESS...
2559 1 ; ...FOR THIS SECTOR NUMBER.
2560 BVC 1$ ; BR IF SECTOR FOUND.
2561 MOVB TSECCX,EM119X ; SET ASCII SECTOR NUMBER.
2562 DFERR EM119,NOSIG ; CAN'T FIND SECTOR.
2563
2564 TRAP C$ERDF
2565 .WORD 171
2566 .WORD EM119
2567 .WORD NOSIG
2568
2569 000004 003312 1$: BR 7$
2570 BIT #BIT2,MPXCLK ; NOMINAL CLOCK ENABLED ??
2571 BNE 2$ ; BR IF NOT.
2572 MOV #224,(R5) ; YES, SET NOMINAL CLOCK, NO PEAK SHIFT.
2573 MOV #TCLK0,TMP2
2574 BR 4$
2575 000002 003312 2$: BIT #BIT1,MPXCLK ; FAST CLOCK ENABLED ??
2576 BNE 3$ ; BR IF NOT.
2577 MOV #230,(R5) ; YES, SET FAST CLOCK, NO PEAK SHIFT.
2578 MOV #TCLK1,TMP2
2579 BR 4$
2580 000001 003312 3$: BIT #BIT0,MPXCLK ; SLOW CLOCK ENABLED ??
2581 BNE 7$ ; BR IF NOT.
2582 MOV #220,(R5) ; YES, SET SLOW CLOCK, NO PEAK SHIFT.
2583 MOV #TCLK2,TMP2
2584
2585 040000 003312 4$: BIT #BIT14,MPXCLK ; MIN PEAK SHIFT ENABLED ??
2586 BNE 5$ ; BR IF NOT.
2587 CLRB TPEAK ; YES, PEAK SHIFT IS OFF...
2588 JSR PC,READSEC ; ...READ SECTOR AND VERIFY DATA.
2589 TST MPXCLK ; MAX PEAK SHIFT ENABLED ??
2590 BMI 6$ ; BR IF NOT.
2591 BIS #40,(R5) ; YES, MAX PEAK SHIFT ON...
2592 MOV #40,TPEAK ; ...EXTEND CLOCK MESSAGE...
2593 JSR PC,READSEC ; ...DO IT ALL AGAIN.
2594
2595 000004 036602 5$: TST MPXCLK
2596 BMI 6$
2597 BIS #40,TPEAK
2598 JSR PC,READSEC
2599
2600 000004 6$: BIT #4,(R5) ; NOMINAL DONE ??
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 H 8
34 -- READ DATA, READ HEADER, AND READ DATA W/O HEADER.

SEQ 0098

```
2586 022732 00330      BNE 2$      ; LOOP TO 'FAST' IF SO.
2587 022734 032715 000010 BIT #10,(R5) ; FAST DONE ??
2588 022740 001337      BNE 3$      ; LOOP TO 'SLOW' IF SO
2589      ; ...OTHERWISE, THIS SECTOR IS DONE.
2590
2591 022742 032737 000007 003314 7$: BIT #7,SNGLSEC ; SINGLE SECTOR MODE ??
2592 022750 001004      BNE 8$      ; WE'RE DONE IF SO.
2593 022752 023727 022544 000006 CMP 12$,#6 ; NO, LAST SECTOR DONE ??
2594 022760 103665      BLO 10$     ; GO 'ROUND AGAIN IF NOT...
2595 022762      8$: EXIT TST ; ...OTHERWISE, NEXT TEST.
2596      ; TRAP C$EXIT
2597      ; .WORD L10061-.
2598
2599      ; SUBROUTINE TO READ DATA FROM TLM TO BUF2 AND VERIFY (AGAINST BUF1).
2600      ; DISK ADDRESS AND EXPECTED DATA (BUF1) MUST BE PRESET BEFORE CALL.
2601      ; WORD COUNT (-128.) IS PRESET FOR ALL.
2602      READSEC: BGNSEG
2603      (3) 022766 104404      TRAP C$BSEG
2604 022770 012700 005364      MOV #BUF2,R0
2605 022774 012701 000200      MOV #128,R1
2606 023000 012720 177777 1$: MOV #-1,(R0)+ ; CLEAR RECEIVING BUFFER.
2607 023004 077103      SOB R1,1$
2608 023006 105037 036205      CLRB EM115X ; INIT ERROR TEXT FOR 'READ'.
2609 023012 105037 036355      CLRB EM117X
2610 023016 012777 000214 160300 MOV #CRDY!READ,ARLCS ; SET COMMAND TO READ...
2611 023024 013777 003436 160300 MOV INIMP,ARLMP ; ...128 WORDS...
2612 023032 013777 003440 160270 MOV INIDA,ARLDA ; ...FROM CYL X, SECT Y...
2613 023040 012777 005364 160260 MOV #BUF2,ARLBA ; ...TO BUF2.
2614 023046 004537 010016 2$: JSR R5,BEFORE
2615 023052 042777 000200 160244 BIC #CRDY,ARLCS ; XCT.
2616 023060 004537 010040 JSR R5,WTCRDY ; WAIT FOR RL DONE.
2617 023064 123737 024506 003365 CMPB PSERR,E.CS+1 ; FINAL STATUS RIGHT ??
2618 023072 001412 BEQ 3$ ; BR IF SO.
2619 023074 004537 007510 JSR R5,GETERR ; NO, GET WHATEVER'S THERE.
2620 023100 000240 NOP
2621 023102 013737 024504 003426 MOV PSETXT,TMP1
2622 023108 013737 024504 003426 DFERR EM115,ERR20 ; ERROR STATUS WRONG ON READ.
2623      ; TRAP C$ERDF
2624      ; .WORD 172
2625      ; .WORD EM115
2626      ; .WORD ERR20
2627 023110 000240
2628 023112 000240
2629 023114 000240
2630 023116 000240
2631 023118 000240
2632 023120 000240
2633 023122 000240
2634 023124 000240
2635 023126 000240
2636 023128 000240
2637 023130 000240
2638 023132 000240
2639 023134 000240
2640 023136 000240
2641 023138 000240
2642 023140 000240
2643 023142 000240
2644 023144 000240
2645 023146 000240
2646 023148 000240
2647 023150 000240
2648 023152 000240
2649 023154 000240
2650 023156 000240
2651 023158 000240
2652 023200 000240
2653 023202 000240
2654 023204 000240
```

CVRLBAO -- RLV12 DISKLESS,
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1053) 08-FEB-82 10:07 PAGE 43-2
34 -- READ DATA, READ HEADER, AND READ DATA W/O HEADER.

SEQ 0099

```
(5) 023204 104455
(6) 023206 000255
(6) 023210 036233
(6) 023212 026742
2635 023214 000460
2636 023216 012777 000216 160100 3$: BR READDUN ; ABORT THE REST.
2637 023218 013777 003436 160100 MOV #CRDY!RDNHDR,@RLCS ; OK, SET READ-NO-HEADER COMMAND...
2638 023220 004537 010016 JSR R5,BEFORE ;...INSURE WORD COUNT IS STILL THERE...
2639 023222 042777 000200 160060 BIC #CRDY,@RLCS ;...AND XCT IT.
2640 023224 004537 010640 JSR R5,WTCDY
2641 023226 004537 007510 JSR R5,GETERR ; SHOULD HAVE NO ERRORS...
2642 023228 000407 BR CHKDATA ;...BR IF SO.
2643 023230 012737 036670 003426 MOV #EXPNOX.TMP1
2644 023232 012737 036670 003426 DFER" EM115,ERR20 ; ERROR STATUS INCORRECT AFTER READ NH
(5) 023264 104455
(6) 023266 000256
(6) 023270 036137
(6) 023272 026662
2645 023274 012700 004444
2646 023276 012701 005364
2647 023278 005002
2648 023280 021011
2649 023282 001415
2650 023284 011037 003432
2651 023286 011137 003434
2652 023288 010137 003424
2653 023290 010237 003426
2654 023292 010237 003426
2655 023294 010237 003426
(5) 023332 104455
(6) 023334 000257
(6) 023336 036313
(6) 023338 026752
2656 023340 000405
2657 023342 022021
2658 023344 005202
2659 023346 032702 000177
2660 023348 001354
2661 023350 001354
2662 023352 001354
(3) 023354 104405
(3) 023356 000207
2663 023358 000207
2664 023360 000207
(3) 023362 104401
(3) 023364 104401

CHKDATA: MOV #BUF1+48.,R0 ; EXPD DATA FIELD BEGINS AT WORD 24...
MOV #BUF2,R1 ;...RECVD DATA AT WORD 0.
CLR R2 ;...AND WORD NUMBER.
1$: CMP (R0),(R1) ; COMPARE EXP'D VS REC'D DATA...
BEQ 2$ ;...AND BR IF OK.
MOV (R0),GDDAT ; IT'S NOT, SET EXP'D...
MOV (R1),BDDAT ;...REC'D...
MOV R1,TMP0 ;...ADDRESS...
MOV R2,TMP1 ;...AND WORD NUMBER.
DFERR EM117,ERR22 ; DATA INCORRECT ON READ OR READ NH.
(5) 023382 104455
(6) 023384 000257
(6) 023386 036313
(6) 023388 026752
2656 023390 000405
2657 023392 022021
2658 023394 005202
2659 023396 032702 000177
2660 023398 001354
2661 023400 001354
2662 023402 001354
(3) 023404 104405
(3) 023406 000207
2663 023408 000207
2664 023410 000207
(3) 023412 104401
(3) 023414 104401

2$: BR 3$
CMP (R0)+,(R1)+ ; BUMP ADDRESSES...
INC R2 ;...AND WORD NUMBER.
BIT #177,R2
BNE 1$ ;...AND LOOP 'TIL DONE.

3$: READDUN: ENDSEG

RTS PC
ENDTST

10000$: TRAP C$ESEG
L10061: TRAP C$ETST
```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 J 8
35 -- WRITE CHECK.

SEQ 0100

```
2666 .SBTTL 35 -- WRITE CHECK.
2667
2668 023364 STARS
2669 (2) :*****
2670 023364 : DO A WRITE CHECK USING SECTOR AND CLOCK OPTIONS AS BEFORE.
2671 (2) STARS
2672 023364 :*****
2673 (4) TLM6: BEGIN.TEST
2674 023364 T35::
2675 023364 004737 023766 JSR PC,TLMPID
2676 023364 102525 BVS 8$ ; EXIT IF NOT OK.
2677 023364 012737 177600 003436 MOV #-128,INIMP ; INIT WORD COUNT.
2678 023364 013737 003314 023426 MOV SNGLSEC,12$ ; GET SECTOR NUMBER (IF ANY).
2679 023364 042737 177770 023426 BIC #^C7,12$
2680 023364 001002 BNE 11$ ; NZ = SINGLE SECTOR MODE.
2681 023364 005237 023426 10$: INC 12$ ; OTHERWISE, USE ALL STARTING WITH 1.
2682 023364 004737 024214 11$: JSR PC,PSX ; SET BUFFERS AND DISK ADDRESS...
2683 023364 000001 12$: 1 ; ...FOR THIS SECTOR NUMBER.
2684 023364 102010 BVC 1$
2685 023364 113737 036532 036521 MOVB TSECX,EM119X ; SET ASCII SECTOR NUMBER.
2686 023364 104455 DFERR EM119,NOSIG ; CAN'T FIND SECTOR.
2687 (5)
2688 (6) TRAP CSERDF
2689 (6) .WORD 176
2690 (6) .WORD EM119
2691 (6) .WORD NOSIG
2692 023364 000004 003312 1$: BR 7$
2693 023364 000224 003430 BNE 2$ ; NOMINAL CLOCK ENABLED ??
2694 023364 000002 003312 2$: BR 4$ ; BR IF NOT.
2695 023364 000230 003430 MOV #224,(R5) ; YES, SET NOMINAL CLOCK, NO PEAK SHIFT.
2696 023364 000001 003312 3$: BR 4$
2697 023364 000220 003430 MOV #230,(R5) ; FAST CLOCK ENABLED ??
2698 023364 000001 003312 3$: BR 4$ ; BR IF NOT.
2699 023364 000220 003430 MOV #220,(R5) ; YES, SET FAST CLOCK, NO PEAK SHIFT.
2700 023364 000001 003312 3$: BR 4$
2701 023364 000220 003430 MOV #220,(R5) ; SLOW CLOCK ENABLED ??
2702 023364 000001 003312 3$: BR 4$ ; BR IF NOT.
2703 023364 000220 003430 MOV #220,(R5) ; YES, SET SLOW CLOCK, NO PEAK SHIFT.
2704 023364 000001 003312 3$: BR 4$
2705 023364 000220 003430 MOV #220,(R5)
2706 023364 000001 003312 3$: BR 4$
2707 023364 000220 003430 MOV #220,(R5)
2708 023364 000001 003312 3$: BR 4$
2709 023364 000220 003430 MOV #220,(R5)
2710 023364 000001 003312 3$: BR 4$
2711 023364 000220 003430 MOV #220,(R5)
2712 023364 000001 003312 3$: BR 4$
2713 023364 000220 003430 MOV #220,(R5)
2714 023364 000001 003312 3$: BR 4$
```

```

2715 023624 032737 000007 003314 7$: BIT #7,SNGLSEC ; SINGLE SECTOR MODE ??
2716 023632 001004 ; WE'RE ALL DONE IF SO.
2717 023634 023727 023426 000006 ; NO, LAST SECTOR DONE ??
2718 023642 103665 ; GO 'ROUND AGAIN IF NOT...
2719 023644 ; ...OTHERWISE, WE'RE ALL DONE.
2720 023644 104432 ;
2721 023646 000112 ; TRAP C$EXIT
2722 ; SUBROUTINE TO WRITE CHECK A SECTOR AGAINST THE DATA IN 'BUF1'. L10062-.
2723 ;
2724 023650 WRITCHK: BGNSEG
2725 023650 104404 ; TRAP C$BSEG
2726 023652 012777 000202 157444 MOV #CRDY!WRCHK,@RLCS ; SET COMMAND TO WRITE-CHECK...
2727 023660 013777 003436 157444 MOV INIMP,@RLMP ; ...128 WORDS...
2728 023666 013777 003440 157434 MOV INIDA,@RLDA ; ...FROM CYL X, SECT Y...
2729 023674 012777 004444 157424 MOV #BUF1+48,@RLBA ; ...AGAINST BUF1.
2730 023702 004537 010016 JSR R5,BEFORE
2731 023706 042537 000200 157410 BIC #CRDY,@RLCS ; XCT.
2732 023714 004537 010640 JSR R5,WTCDY ; WAIT FOR RL DONE.
2733 023720 123757 024506 003365 CMPB PSERR,E.CS+1 ; FINAL STATUS RIGHT ??
2734 023726 001416 BEQ 1$ ; BR IF SO.
2735 023730 004537 007510 JSR R5,GETERR ; NO, GET WHATEVER'S THERE.
2736 023734 000240 NOP
2737 023736 013737 024504 003426 MOV PSETXT,TMP1
2738 023744 104455 DFERR EM118,ERR20 ; ERROR STATUS WRONG ON WRITE-CHECK
2739 023746 000261 ; TRAP C$ERDF
2740 023750 036403 ; .WORD 177
2741 023752 026662 ; .WORD EM118
2742 023754 ; .WORD ERR20
2743 023754 104405 1$: ENDSEG
2744 023756 000207 ; 10000$: TRAP C$ESEG
2745 023760 ;
2746 023760 104401 ; L10062: TRAP C$ETST
2747 023760 ;

```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 306(1063) 08-FEB-82 10:07 L 8
35 -- WR. 2 CHECK. PAGE 44-2

SEQ 0102

```
2742  
2743  
2744  
2745  
2746  
2747 023762 005000  
2748 023764 000401  
2749 023766 005200  
2750 023770 013737 003344 003432  
2751 023776 001453  
2752 024000 005037 010626  
2753 024004 005077 157334  
2754 024010 000240  
2755 024012 005737 010626  
2756 024016 001025  
2757 024020 012737 004400 003432  
2758 024026 017737 157312 003434  
2759 024034 001423  
2760 024036 005700  
2761 024040 001413  
2762 024042 004737 024132  
2763 024046 010037 003434  
2764 024052 013737 003310 003432  
2765 024060 023737 003432 003434  
2766 024066 001013  
2767 024070 000207  
2768  
2769 024072  
(5) 024072 104455  
(6) 024074 000262  
(6) 024076 034574  
(6) 024100 026170  
2770 024102 000411  
2771 024104  
(5) 024104 104455  
(6) 024106 000263  
(6) 024110 034630  
(6) 024112 026220  
2772 024114 000404  
2773 024116  
(5) 024116 104455  
(6) 024120 000264  
(6) 024122 034713  
(6) 024124 026220  
2774 024126 000262  
2775 024130 000207  
2776  
2777  
2778  
2779  
2780  
2781  
2782 024132 013705 003344  
2783 024136 012715 000701  
2784 024142 016500 000006  
2785 024146 005065 000002
```

```

: SUBROUTINE TO SEE IF TLM IS THERE AND PROPERLY CABLED TO RLV12.
: ALSO CHECK THAT CORRECT PROM IS INSTALLED.
: REPORT ERROR AND RETURN WITH 'V' SET IF NOT.
TLMOK: CLR R0 ; DISABLE ID CHECK.
        SKP1
TLMPID: INC R0 ; ENABLE ID CHECK.
        MOV TCSR,GDDAT ; GET CSR POINTER.
        BEQ 4$ ; ABORT IF IT'S NOT DEFINED.
        CLR TRPFLG
        CLR @TCSR ; TLM ARE YOU REALLY THERE ??
        240
        TST TRPFLG
        BNE 1$ ; ERROR IF NOT.
        MOV #4400,GDDAT ; OTHERWISE
        MOV @TCSR,BDDAT ; ...GET INITIAL STATUS.
        BEQ 2$ ; IF ZERO, CABLE ISN'T CONNECTED.
        TST R0 ; ID VERIFICATION REQUIRED ??
        BEQ 5$ ; WE'RE DONE IF NOT.
        JSR PC,CLRTLM ; YES, RESET, ID => R0.
        MOV R0,BDDAT
        MOV PROMID,GDDAT ; GET EXPECTED ID.
        CMP GDDAT,BDDAT
        BNE 3$ ; BR IF ID INCORRECT.
        RTS PC ; RETURN, 'V' = 0.
5$:
1$: DFERR EM100,ERR1 ; TIME-OUT ON TLM ADDRESS.
        TRAP .WORD CSERDF
        .WORD 178
        .WORD EM100
        .WORD ERR1
2$: BR 4$
        DFERR EM100A,ERR2 ; CABLE PROBABLY NOT INSTALLED.
        TRAP .WORD CSERDF
        .WORD 179
        .WORD EM100A
        .WORD ERR2
3$: BR 4$
        DFERR EM100B,ERR2 ; PROM ID INCORRECT.
        TRAP .WORD CSERDF
        .WORD 180
        .WORD EM100B
        .WORD ERR2
4$: SEV ; SET 'ERROR' FLAG...
        RTS PC ; ...AND RETURN.

: SUBROUTINE TO RESET TEST LOOP MODULE.
: RETURN WITH 'DRIVE READY' SET, TLM PROM ID IN R0,
: AND TLM CSR ADDRESS IN R5.
: AS LONG AS WE'RE HERE, CLEAR RLBA, RLMP, AND RLBAE AS WELL.
CLRTLM: MOV TCSR,R5 ; DEDICATE R5 TO THE TLM.
        MOV #1,(R5) ; RESET...
        MOV 6(R5),R0 ; ...PROM ID => R0.
        CLR 2(R5) ; CLEAR SKGS...
```



```

2786 024152 005065 000004 CIR 4(R5) ;...AND PSUEDO-STATUS REGISTERS.
2787 024156 012715 000002 MOV #2(R5) ; CLEAR FLAGS ??
2788 024162 012715 000200 MOV #200(R5) ; SET DRIVE READY.
2789 024166 005077 157134 CLR @RLBA
2790 024172 005077 157134 CLR @RLMP
2791 024176 023727 003402 000003 CMP R1YP,#RLV12X
2792 024204 001002 157122 BNE 1$
2793 024206 005077 157122 CLR @RLBAE
2794 024212 000207 1$ RTS PC
2795
2796 ; SUBROUTINE TO SET UP BUFFERS AND DISK ADDRESSES
2797 ; FOR READING EACH OF THE 6 PSUEDO SECTORS.
2798 ; USE HEADER TABLE AND SEARCH ROUTINE IN THE DUMP UTILITY.
2799 ; IF SUCCESSFUL, CONTENTS OF SELECTED SECTOR (160. WORDS) ARE IN 'BUF1'.
2800 CALL: JSR PC,PSX
2801 N ; SECTOR NUMBER.
2802 BVC XX ; ON RETURN, 'V' IS SET IN SECTOR NOT FOUND.
2803 ERROR
2804 XX: ; CONTINUE.
2805
2806 024214 017600 000000 PSX: MOV @ (SP),R0 ; GET SECTOR NUMBER (1 - 6)...
2807 024220 005300 DEC R0 ;...MAKE IT (0 - 5)...
2808 024222 006300 ASL R0 ;...SHIFT UP TO AN INDEX.
2809 024224 062716 000002 ADD #2(SP) ; ADJUST RETURN PC...
2810 024230 000170 024234 JMP @1$(R0) ;...AND DO ONE OF THE FOLLOWING:
2811 024234 C24252 1$ PS1
2812 024236 C24270 PS2
2813 024240 C24322 PS3
2814 024242 C24354 PS4
2815 024244 C24372 PS5
2816 024246 C24442 PS6
2817 024250 024410 PS7
2818 ; SOONER OR LATER SOME TURKEY WILL...
2819 ;...TRY TO ACCESS SECTOR 7.
2820
2821 024252 112737 000061 036532 PS1: .ENABL LSB
2822 024260 012737 000205 003440 MOVB #1,TSECX ; TLM SECTOR 1...
2823 024266 000473 MOV #<1_7>!5,INIDA ;...IS CYL 1 SECT 5...
2824 BR 1$ ;...AND IS ERROR FREE.
2825
2826 024270 112737 000062 036532 PS2: MOVB #2,TSECX ; TLM SECTOR 2...
2827 024276 012737 000206 003440 MOV #<1_7>!6,INIDA ;...IS CYL 1, SECT 6...
2828 024304 012737 000210 024506 MOV #^H^ZERR!D^CRC>,PSERR
2829 024312 012737 036650 024504 MOV #EXPDCRC,PSETXT ;...AND HAS A BAD DATA CRC.
2830 024320 000463 BR 2$
2831
2832 024322 112737 000063 036532 PS3: MOVB #3,TSECX ; TLM SECTOR 3...
2833 024330 012737 000207 003440 MOV #<1_7>!7,INIDA ;...IS CYL 1, SECT 7...
2834 024336 012737 000214 024506 MOV #^H^ZERR!H^CRC!OPI>,PSERR
2835 024344 012737 036624 024504 MOV #EXPHCRC,PSETXT ;...AND HAS A BAD HEADER CRC.
2836 024352 000446 BR 2$
2837
2838 024354 112737 000064 036532 PS4: MOVB #4,TSECX ; TLM SECTOR 4...
2839 024362 012737 125252 003440 MOV #<525_7>!52,INIDA ;...IS CYL 525, SECT 52...
2840 024370 000432 BR 1$ ;...AND IS ERROR FREE.
2841 024372 112737 000065 036532 PS5: MOVB #5,TSECX ; TLM SECTOR 5...

```

CVRLBAO -- RLV12 DISKLESS MACY11 30G(1063) 08-FEB-82 10:07 N 8
CVRLBA.P11 08-FEB-82 09:59 35 -- WRITE CHECK. PAGE 44-4

SEQ 0104

```
2842 024400 012737 135252 003440      MOV    #<565_7.>!52,INIDA ;...IS CYL 565, SECT 52...
2843 024406 000423                BR      18 ;...AND IS ERROR FREE.
2844
2845 024410                PS7: PRINTF #TURKY ; THERE ISN'T ANY SECTOR 7...
(7) 024410 012746 024510                MOV    #TURKY-(SP)
(6) 024414 012746 000001                MOV    #1, -(SP)
(3) 024420 010670                MOV    SP, RO
(4) 024422 104417                TRAP   C$PNTF
(4) 024424 062706 000004                ADD    #4, SP
2846 024430 012737 000006 003314      MOV    #6, SNGLSEC ;...CHANGE IT TO 6...
2847 024436 162700 000002      SUB     #2, RO ;...ADJUST INDEX AND FALL THRU.
2848
2849 024442 112737 000066 036532 PS6: MOVB   #'6, TSECX ; TLM SECTOR 6...
2850 024450 012737 155555 003440      MOV    #<666_7.>!155,INIDA ;...IS CYL 666, SECT 55, HEAD 1...
2851 024456 005037 024506      18:    CLR    PSERR
2852 024462 012737 036670 024504      MOV    #EXPNON, PSETXT
2853 024470 016037 024572 024570 28:    MOV    DIABL(RO), DHEAU ; SET HCRC TO SEARCH FOR.
2854 024476 004737 025174      JSR     PC, FINDSEC ; GO FIND AND READ IT TO BUF1.
2855 024502 000207                RTS     PC ; RETURN, 'V' = 1 IF SECTOR NOT FOUND.
2856                .DSABL LCB
2857
2858 024504 036670                PSETXT: EXPNON ; EXPECTED ERROR STATUS FOR EACH...
2859 024506 000000                PSERR: 0 ;...IS RETURNED IN THESE LOCATIONS.
2860
2861 024510 047045 040445 042523 TURKY: .ASCIZ /%N%ASECTOR 7 DOESN'T EXIST, USING 6 INSTEAD%N/
2862                .EVEN
```


2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919

024566	025324	
024570	006351	
024572	006051	
024574	044051	
024576	132051	
024600	174001	
024602	036400	
024604	113554	
024606	005037	003306
024612	005000	
024614	077001	
024616	104005	
024620	113702	025701
024624	004737	025476
024630	025561	
024632	004737	025432
024636	113700	004364
024642	001407	
024644	120027	000060
024650	103763	
024652	120027	000067
024656	101360	
024660	110002	
024662	110201	
024664	013705	003320
024670	042701	177770
024674	001403	
024676	062705	000010
024702	077103	
024704	110237	025701
024710	012737	024726
024716	012715	000001
024722	000240	
024724	000410	

000004

```

.SBTTL          TLM PROM DUMP UTILITY.

ROUTINE TO DUMP TLM ROM CONTENTS.
CALLED FROM "INIT" SECTION IF PDSW=1.
GET UNIT NUMBER, SECTOR NUMBER, AND OUTPUT FORMAT FROM OPERATOR.
<^C> = ABORT AND RESTART DRS.
LOAD SUBROUTINE USES "BUF2" AS A 256. WORD RING BUFFER FOR SEARCHING.
IF FOUND, SECTOR CONTENTS (160. WORDS) ARE THEN COPIED TO "BUF1".
THE LOAD SUBROUTINE IS ALSO USED IN TLM TESTS 5 AND 3
TO ESTABLISH EXPECTED DATA PATTERNS.

PSUEDO-DISK (ROM) SECTOR FORMAT:
      WORDS          FUNCTION
      -----
      1 - 17         SECTOR PULSE (12), T3(2), HEADER PREAMBLE (3).
      18 - 20        DISK ADDRESS (1), ZERO (1), HEADER CRC (1).
      21 - 24        HEADER POSTAMBLE (1), DATA PREAMBLE (3).
      25 - 153       DATA (128.), DATA CRC (1).
      154 - 160      DATA POSTAMBLE (1), AND GAP (6).

DMODE: OCTMOD      : DUMP MODE (OCTAL OR HEX).
DHEAD: 006051      : DESIRED HEADER CRC WORD (DEFAULT SECT 1)...
DTABL: 006051      : ...SELECTED FROM THIS TABLE.
      044051      : 0C29
      132051      : 4829
      174001      : B429
      036400      : F801
      113554      : 3D00
      113554      : 976C

ROMDUMP: CLR        PDSW          : CLEAR DUMP REQUEST.
          CLR        R0
          SOB        R0,,
          EMT+5      : DELAY...
          MOV B      HUNIT,R2      : ...AND CLEAR KBD.
          JSR        PC,KBDOUT    : GET DEFAULT UNIT.
          R0          : SAY "HELLO".
          JSR        PC,KBDIN      : GET A UNIT NUMBER.
          MOV B      BUF1,R0
          BEQ        2$           : IF NONE, USE DEFAULT (LAST).
          CMP B      R0,#'0
          BLO        1$           : BR IF INVALID UNIT.
          CMP B      R0,#'7
          BHI        1$           :
          MOV B      R0,R2        : DITTO
          MOV B      R2,R1        : SAVE NEW UNIT FOR A MOMENT.
          MOV        R2,R1
          MOV        TC50,R5      : BASE CSR => R5.
          BIC        #'C7,R1     : STRIP UNIT NUM...
          BEQ        3$          : ...AND BR IF ZERO.
          ADD        #'10,R5      : ELSE, ADJUST UP TO SELECTED UNIT.
          SOB        R1,22$

          1$:
          2$:
          3$: MOV B      R2,HUNIT   : OK, SET UNIT IN TEXT.
              MOV        #'4$ ,ERRVEC : SET A TRAP CATCHER.
              MOV        #'1,(R5)     : RESET TLM...
              240
              BR         5$          : ...AND PROCEED IF IT DIDN'T TRAP.

```

```

2920 024726 022628 025701 025536 48: CMP (SP)+,(SP)+
2921 024730 113737 MOVH HUNIT,NOANS+6
2922 024736 004737 JSR PC,KBDOUT ; SELECTED UNIT DOESN'T ANSWER.
2923 024742 02553C NOANS
2924 024744 000720 BR ROMDUMP
2925
2926 024746 004737 02547J 58: JSR PC,KBDOUT ; GET A SECTOR NUMBER.
2927 024752 025612 Q1
2928 024754 004737 JSR PC,KBDIN
2929 024760 113700 MOVH BUF1,R0
2930 024764 001416 BEQ 68 ; IF NONE, REDO LAST SECTOR.
2931 024766 120027 000060 CMPB R0,#'0
2932 024772 103765 BLO 58
2933 024774 120027 000067 CMPB R0,#'7
2934 025000 101362 BHI 58
2935 025002 110037 025714 MOVH R0,HSECT ; SET IN HEADER STRING.
2936 025006 042700 177770 BIC #^C7,R0
2937 025012 006300 ASL R0 ; SHIFT UP TO WORD OFFSET...
2938 025014 016037 024570 024570 MOV DHEAD(R0),DHEAD ;...AND SET HEADER TO SEARCH FOR.
2939
2940 025022 004737 025476 68: JSR PC,KBDOUT ; GET AN OUTPUT MODE.
2941 025026 025637 Q2
2942 025030 004737 JSR PC,KBDIN
2943 025034 113700 MOVH BUF1,R0
2944 025040 001411 BEQ 78 ; IF NONE, USE CURRENT FORMAT.
2945 025042 012737 025324 024566 MOV #OCTMOD,DMODE ; ASSUME OCTAL...
2946 025050 120027 000131 CMPB R0,#'Y
2947 025054 001003 BNE 78
2948 025056 012737 025350 024566 MOV #HEXMOD,DMODE ;...CHANGE TO HEX IF REQUIRED.
2949
2950 025064 016537 000006 003424 78: DOIT: MOV 6(R5),TMP0 ; GET PROM ID.
2951 025072 004737 JSR PC,KBDOUT ; SHOW UNIT AND SECTOR...
2952 025076 025670 HEADR
2953 025100 012701 003424 MOV #TMP0,R1 ;...AND PROM ID...
2954 025104 004777 177456 JSR PC,@DMODE ;...IN THE CURRENT CONTEXT.
2955 025110 104007 EMT+7 ; <CRLF>
2956 025112 004737 025174 JSR PC,FINDSEC ; SEARCH FOR AND READ SECTOR.
2957 025116 102004 BVC 18 ; BR IF SECTOR FOUND.
2958 025120 004737 025476 JSR PC,KBDOUT
2959 025124 025731 CANTFIND
2960 025126 000627 BR ROMDUMP
2961
2962 025130 012701 004364 18: MOV #BUF1,R1 ; NOW FOUND SECTOR IS IN BUF1.
2963 025134 012702 000240 MOV #160.,R2 ; DUMP 160. WORDS AT 8 PER LINE.
2964 025138 104007 EMT+7 ; <CRLF>
2965 025142 004737 025476 28: JSR PC,KBDOUT
2966 025146 025737 TAB
2967 025150 004777 177412 JSR PC,@DMODE ; PRINT FROM (R1) IN OCTAL OR HEX.
2968 025154 003702 DEC R2
2969 025158 032702 000007 BIT #7,R2
2970 025162 001369 BNE 28 ; LOOP 'TIL LINE DONE...
2971 025166 104007 EMT+7 ; <CRLF>
2972 025168 005702 TST R2
2973 025170 001364 BNE 28 ;...AND LOOP 'TIL ALL DONE.
2974 025172 000605 BR ROMDUMP
2975

```

```

2976
2977
2978
2979 025174 012715 000001
2980 025200 012715 000034
2981 025204 005046
2982 025206 005000
2983 025210 012701 006364
2984 025214 020127 005364
2985 025220 001002
2986 025222 012701 006364
2987 025226 005265 000006
2988 025232 016541 000006
2989 025236 021137 024570
2990 025242 001003
2991 025244 005216
2992 025246 012700 000024
2993 025252 077020
2994 025254 012715 000001
2995 025260 012715 000200
2996 025264 005726
2997 025266 001002
2998 025270 00J262
2999 025272 000207
3000
3001 025274 012702 004364
3002 025300 012703 000240
3003 025304 020127 006364
3004 025310 001002
3005 025312 012701 005364
3006 025316 012122
3007 025320 077307
3008 025322 000207
3009
3010
3011
3012 025324 012146
3013 025326 012746 025770
3014 025332 012746 000002
3015 025336 010600
3016 025340 104417
3017 025342 062706 000006
3018 025346 000207
3019
3020 025350 010546
3021 025352 012700 025763
3022 025356 012104
3023 025360 005005
3024 025362 012703 000004
3025 025364 006304
3026 025370 006105
3027 025372 077303
3028 025374 062705 000060
3029 025400 120527 000071
3030 025404 003402

```

; SEARCH FOR SELECTED SECTOR AND LOAD 160, WORD BLOCK TO BUF1.
; IF UNSUCCESSFUL, RETURN TO CALLER WITH 'V' BIT SET.

```

FINDSEC: MOV #1,(R5) ; RESET TLM...
          MOV #34,(R5) ; ...AND SET PROM READ MODE.
          CLR -(SP) ; CLEAR 'FOUND' FLAG.
          CLR R0 ; SET LOOP CONTROL...
          MOV #BUFEND,R1 ; NOW, FILL BUF2 FROM TOP DOWN.
1$: CMP R1,#BUF2 ; IN LOOP -- RESET RING POINTER...
     BNE 2$ ; ...WHEN IT REACHES THE BOTTOM.
          MOV #BU'END,R1
2$: INC 6(R5) ; INCR ROM ADDRESS...
     MCV 6(R5),-(R1) ; ...AND GET A WORD.
     CMP (R1),DHEAD ; IS THIS THE DESIRED MCRC WORD ??
     BNE 3$ ; BR IF NOT.
     INC (SP) ; YES, SET 'FOUND' FLAG...
          MOV #20,R0 ; ...AND SET TO READ 19 MORE.
3$: SOB R0,1$ ; LOOP 'TIL DONE...
     MOV #1,(R5) ; ...THEN, RESET TLM...
     MOV #200,(R5) ; ...AND RESTORE DRIVE READY.
     TST (SP)+ ; SECTOR FOUND ??
     BNE 4$ ; BR IF SO.
     SEV ; NO, SET 'V' BIT...
     RTS PC ; ...AND RETURN.

4$: MOV #BUF1,R2 ; NOW COPY FOUND SECTOR TO BUF1.
     MOV #160,R3
5$: CMP R1,#BUFEND ; IN LOOP -- RESET RING POINTER...
     BNE 6$ ; ...WHEN IT REACHES THE TOP.
     MOV #BUF2,R1
6$: MOV (R1)+,(R2)+ ; BUF2 => BUF1.
     SOB R3,5$ ; RETURN, 'V' IS CLEAR.
     RTS PC

```

; SUBROUTINES TO DUMP DATA IN OCTAL OR HEX.

OCTMOD: PRINTF #OCTTXT,(R1)+

```

MOV (R1)+,-(SP)
MOV #OCTTXT,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PRINTF
ADD #6,SP

```

RTS PC

```

HEXMOD: MOV R5,-(SP) ; SAVE TLM POINTER.
1$: MOV #HEXTXT,R0 ; SET HEX TEXT POINTER.
     MOV (R1)+,R4 ; NEXT WORD => R4.
2$: CLR R5 ; ...AND PARSE EACH NIBBLE IN R5.
     MOV #4,R3
3$: ASL R4
     ROL R5
     SOB R3,3$ ; NIBBLE => R5<3:0>
     ADD #10,R5 ; CONVERT TO ASCII 0 TO 9...
     CMPB R5,#9
     BLE 4$

```

```

3026 025406 062705 000007      ADD    #7,95      ;...AND A(10) TO F(15)...
3027 025412 110520      4$:  MOV    R5,(R0)+    ;...AND STUFF IT.
3028 025414 105710      TSTB    (R0)
3029 025416 001360      BNE     2$      ; LOOP 'TIL 4 CHARS ASSEMBLED...
3030 025420 004737      JSR     PC,KBDOUT    ;...THEN DUMP IT.
3031 025424 035763      HEXTXT
3032 025426 012605      MOV    (SP)+,R5    ; RESTORE R5.
3033 025430 000207      RTS     PC
3034 025432 012701 004364      KBDIN: MOV    #BUF1,R1    ; TEMP ASCII BUFFER.
3035 025436 104005      1$:  EMT+5    ; CHAR => R0.
3036 025440 110011      MOV    R0,(R1)
3037 025442 001775      BEQ     1$
3038 025444 120027      CMPB    R0,#3
3039 025446 001411      BEQ     3$      ; EXIT ON <^C>.
3040 025450 120027      CMPB    R0,#15
3041 025452 001403      BEQ     2$      ; BR ON TERMINATOR.
3042 025456 104004      EMT+4    ; OTHERWISE, ECHO...
3043 025460 005201      INC     R1      ;...BUMP POINTER...
3044 025462 000764      BR     1$      ;...AND LOOP.
3045 025464 105011      2$:  CLRA    (R1)    ; TERMINATE INPUT STRING...
3046 025466 104007      EMT+7    ;...ECHO CRLF...
3047 025470 000207      RTS     PC      ;...AND RETURN TO CALLER.
3048 025472 104444      3$:  DOCLN    ; EXIT ON <^C>.
3049 025474 104444

```

TRAP C\$DCLN

```

MOV    @ (SP),-(SP)
MOV    #PURE, -(SP)
MOV    #2, -(SP)
MOV    SP,R0
TRAP   C$PNTF
ADD     #6,SP

```

```

3050 025476 017646 000000      KBDOUT: PRINTF #PURE,@(SP)
3051 025478 012746 025774
3052 025480 012746 000002
3053 025482 010600
3054 025484 104417
3055 025486 062706 000006
3056 025488 062716 000002
3057 025490 000207
3058 025530 020040 046124 035115      NOANS: .ASCII / TLM:0 DOESN'T ANSWER/<15><12>
3059 025532 015 012      Q0: .ASCII <15><12>
3060 025534 040 052040 046514      .ASCII / TLM UNIT (0 TO 7) ? /
3061 025536 020040 042523 052103      Q1: .ASCII / SECTOR (1 TO 6) ? /
3062 025538 040 044040 054105      Q2: .ASCII / HEX OUTPUT (Y OR N) ? /
3063 025540 005015 052411 044516      HEADR: .ASCII <15><12><11>/UNIT: /
3064 025542 060 020040 042523      HUNIT: .ASCII /0 SECTOR: /
3065 025544 020061 050040 047522      HSECT: .ASCII /1 PROM ID: /
3066 025546 011 040503 023516      CANTFIND: .ASCII <11>/CAN'T FIND HEADER/<15><12>
3067 025548 000011      TAB: .ASCII <11>
3068 025550 005015 000      CRLF: .ASCII <15><12>
3069 025552 101 041502 000104      HEXTXT: .ASCII /ABCD/
3070 025554 047445 000066      OCTTXT: .ASCII /X06/
3071 025556 052045 000      PURE: .ASCII /XT/
3072 025558 026000      .EVEN

```

```

3071
3072
3073
3074
3075
3076 026000 013700 003402
3077 026004 001414
3078 026006 020027 000002
3079 026012 103306
3080 026014 005737 003344
3081 026020 001403
3082 026022 012700 000043
3083 026026 000402
3084 026030 012700 000035
3085 026034 000402
3086 026036 012700 000026
3087 026042 010037 026050
3088 026046 000207
3089
3090 026050
(4) 026050 000043
(3) 026052
(6) 026052 012114
(6) 026054 012200
(6) 026056 012264
(6) 026060 012350
(6) 026062 012434
(6) 026064 012556
(6) 026066 013020
(6) 026070 013140
(6) 026072 013244
(6) 026074 013332
(6) 026076 013436
(6) 026100 013664
(5) 026102 014050
(6) 026104 014216
(6) 026106 014410
(6) 026110 014612
(6) 026112 015036
(6) 026114 015262
(6) 026116 015352
(6) 026120 015746
(6) 026122 016160
(6) 026124 016246
(6) 026126 016414
(6) 026130 016550
(6) 026132 016634
(6) 026134 016762
(6) 026136 017662
(6) 026140 020306
(6) 026142 020502
(6) 026144 021276
(6) 026146 021650
(6) 026150 022034
(6) 026152 022222
(6) 026154 022502

```

```

.SBTTL
.SBTTL SUPERVISOR DISPATCH TABLE.
SUBROUTINE TO ADJUST THE NUMBER OF TESTS TO RUN IAW CONTROLLER TYPE.
ADJTN: MOV RLTP,R0 ; GET RL TYPE.
      BEQ 2$ ; BR IF RL11.
      CMP R0,#RLV12
      BLO 1$ ; BR IF RLV11.
      TST TCSR ; BR IF TLM BYPASSED.
      BEQ 1$ ; RLV12 WITH TLM RUNS ALL.
      MOV #35.,R0 ; RLV12 OR RLV11 RUN THRU 29.
1$: SKP2 #29.,R0 ; RLV12 OR RLV11 RUN THRU 29.
2$: MOV #22.,R0 ; RL11 RUN THRU 22.
      MOV R0,L$DISPATCH-2 ;...AND SET LAST TEST NUMBER.
      RTS PC

```

DISPATCH T\$TESTNUM

```

WORD 35
L$DISPATCH::
WORD T1
WORD T2
WORD T3
WORD T4
WORD T5
WORD T6
WORD T7
WORD T8
WORD T9
WORD T10
WORD T11
WORD T12
WORD T13
WORD T14
WORD T15
WORD T16
WORD T17
WORD T18
WORD T19
WORD T20
WORD T21
WORD T22
WORD T23
WORD T24
WORD T25
WORD T26
WORD T27
WORD T28
WORD T29
WORD T30
WORD T31
WORD T32
WORD T33
WORD T34

```

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07^{6 9} PAGE 45-1
SUPERVISOR DISPATCH TABLE.

SEQ 0110

(6) 026156 023364

.WORD T35

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 H 9
GLOBAL ERROR HANDLERS AND ASCII TEXT

SEQ 0111

```
3092      .SBTTL GLOBAL ERROR HANDLERS AND ASCII TEXT
3093      :
3094      : THESE ARE THE HANDLERS FOR THE VARIOUS ERROR SIGNATURES.
3095      :
3096      BGNMSG ERRSIG
3097      (3) 026160 000137 027306      NOSIG: JMP CKERLT      ; SOME HAVE NO SIGNATURE. ERRSIG::
3098      (3) 026160 000137 027100      ERR0: JMP ALLREGS      ; ALL REGISTERS.
3099      (3) 026164 000137 027100      ERR1: PRINTB #FRMT2,GDDAT      ; BUS-TIMEOUT REG ADDR.
3100      (8) 026170 013746 003432      MOV GDDAT,-(SP)
3101      (7) 026174 012746 027501      MOV #FRMT2,-(SP)
3102      (6) 026200 012746 000002      MOV #2,-(SP)
3103      (3) 026204 010600      MOV SP,R0
3104      (4) 026206 104414      TRAP C$PNTB
3105      (4) 026210 062706 000006      ADD #6,SP
3106      (4) 026214 000137 027306      JMP CKERLT
3107      (9) 026220 013746 003434      ERR2: PRINTB #FRMT4,GDDAT,BDDAT      ; EXP'D VS REC'D.
3108      (8) 026224 013746 003432      MOV BDDAT,-(SP)
3109      (7) 026230 012746 027602      MOV GDDAT,-(SP)
3110      (6) 026234 012746 000003      MOV #FRMT4,-(SP)
3111      (3) 026240 010600      MOV #3,-(SP)
3112      (4) 026242 104414      MOV SP,R0
3113      (4) 026244 062706 000010      TRAP C$PNTB
3114      (4) 026250 000137 027306      ADD #10,SP
3115      (8) 026254 013746 003434      ERR3: PRINTB #FRMT5,BDDAT      ; MAINT SEQ FAILED STATE.
3116      (7) 026260 012746 027636      MOV BDDAT,-(SP)
3117      (6) 026264 012746 000002      MOV #FRMT5,-(SP)
3118      (3) 026270 010600      MOV #2,-(SP)
3119      (4) 026272 104414      MOV SP,R0
3120      (4) 026274 062706 000006      TRAP C$PNTB
3121      (4) 026300 000137 027100      ADD #6,SP
3122      (9) 026304 013746 003434      ERR4: PRINTB #FRMT4,GDDAT,BDDAT      ; EXP'D VS REC'D...
3123      (8) 026310 013746 003432      MOV BDDAT,-(SP)
3124      (7) 026314 012746 027602      MOV GDDAT,-(SP)
3125      (6) 026320 012746 000003      MOV #FRMT4,-(SP)
3126      (3) 026324 010600      MOV #3,-(SP)
3127      (4) 026326 104414      MOV SP,R0
3128      (4) 026330 062706 000010      TRAP C$PNTB
3129      (4) 026334 000137 027100      ADD #10,SP
3130      (9) 026340 013746 034404      ERR6: PRINTB #FRMT3,#_199      ; CS ERRORS.
3131      (8) 026344 012746 027517      MOV #EM99,-(SP)
3132      (6) 026350 012746 000002      MOV #FRMT3,-(SP)
3133      (3) 026354 010600      MOV #2,-(SP)
3134      (4) 026356 104414      MOV SP,R0
3135      (4) 026360 062706 000006      TRAP C$PNTB
3136      (4) 026364 000137 027100      ADD #6,SP
3137      JMP ALLREGS      ;...AND REGISTERS.
```

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 3CG(1063) 08-FEB-82 10:07 PAGE 46-1
GLOBAL ERROR HANDLERS AND ASCII TEXT

SEC 0112

3115
3116 026370
(8) 026370 013746 003424
(7) 026374 012746 027670
(6) 026400 012746 000002
(3) 026404 010600
(4) 026406 104414
(4) 026410 062706 000006
3117 026414 000137 027306
3118
3119 026420
(8) 026420 012746 034404
(7) 026424 012746 027541
(6) 026430 012746 000002
(3) 026434 010600
(4) 026436 104414
(4) 026440 062706 000006
3120 026444 000137 027100
3121
3122 026450
(8) 026450 013746 003434
(7) 026454 012746 027723
(6) 026460 012746 000002
(3) 026464 010600
(4) 026466 104414
(4) 026470 062706 000006
3123 026474 000137 027306
3124
3125 026500 004737 026610
3126 026504 012746 034404
(8) 026504 012746 030241
(7) 026510 012746 000002
(6) 026514 012746 000002
(3) 026520 010600
(4) 026522 104414
(4) 026524 062706 000006
3127 026530 000137 027100
3128
3129 026534 004737 026610
3130 026540 000137 026340
3131
3132 026544 004737 026610
3133 026550 013746 003434
(10) 026550 013746 003432
(9) 026554 013746 003424
(8) 026554 012746 030276
(7) 026560 012746 000004
(6) 026570 010600
(3) 026574 104414
(4) 026576 062706 000012
(4) 026600 000137 027100
3134
3135
3136
3137
3138 026610 005737 003430

ERR7: PRINTB #FRMT6,TMPO ; CPU PRIORITY.

MOV TMPO, -(SP)
MOV #FRMT6, -(SP)
MOV #2, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #6, SP

JMP CKERLT

ERR10: PRINTB #FRMT3A, #EM99 ; EXPECTED OPI ERRORS.

MOV #EM99, -(SP)
MOV #FRMT3A, -(SP)
MOV #2, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #6, SP

JMP ALLREGS

ERR11: PRINTB #FRMT10,BDDAT ; OPI TIMING ERROR.

MOV BDDAT, -(SP)
MOV #FRMT10, -(SP)
MOV #2, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #6, SP

JMP CKERLT

ERR12: JSR PC, PRTPA ; STAT WRONG ON NXM (ABOVE 32K)...
PRINTB #FRMT21, #EM99 ; ...OR BBS7/NXM FAILURE.

MOV #EM99, -(SP)
MOV #FRMT21, -(SP)
MOV #2, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #6, SP

JMP ALLREGS

ERR13: JSR PC, PRTPA ; UNEX STAT ERRORS (ABOVE 32K).
JMP ERR6

ERR14: JSR PC, PRTPA ; DATA WRONG (ABOVE 32K).
PRINTB #FRMT22, TMPO, GDDAT, BDDAT ; EXPD VS RECD.

MOV BDDAT, -(SP)
MOV GDDAT, -(SP)
MOV TMPO, -(SP)
MOV #FRMT22, -(SP)
MOV #4, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #12, SP

JMP ALLREGS

... PACK AND PRINT 22 BIT PHYSICAL ADDRESS FROM TMP1., TMP2.

PRTPA: TST TMP2 ; BIT 15 ON ??


```

3139 026614 100003
3140 026616 002737 100000
3141 026624 006137 003426
3142 026630
(9) 026630 013746 003430
(8) 026634 013746 003426
(7) 026640 012746 030201
(6) 026644 012746 000003
(3) 026650 010600
(4) 026652 104414
(4) 026654 062706 000010
3143 026660 000207
3144
3145
3146
3147 026662 004737 027042
3148 026666
3149 026666
(9) 026666 012746 034404
(8) 026672 013746 003426
(7) 026676 012746 030410
(6) 026702 012746 000003
(3) 026706 010600
(4) 026710 104415
(4) 026712 062706 000010
3150 026716
(7) 026716 012746 030431
(6) 026722 012746 000001
(3) 026726 010600
(4) 026730 104415
(4) 026732 062706 000004
3151 026736 000137 027100
3152
3153 026742 004737 027042
3154 026746 000137 027306
3155
3156 026752 004737 027042
3157 026756
(11) 026756 013746 003434
(10) 026762 013746 003432
(9) 026766 013746 003424
(8) 026772 013746 003426
(7) 026776 012746 030031
(6) 027002 012746 000005
(3) 027006 010600
(4) 027010 104415
(4) 027012 062706 000014
3158 027016
(7) 027016 012746 030431
(6) 027022 012746 000001
(3) 027026 010600
(4) 027030 104415
(4) 027032 062706 000004
3159 027036 000137 027100
3160
3161 027042

```

```

003430 1$: BPL 1$ : NO 'C' = 0;
ADD #BIT15,TMP2 : YES 'C' = 1;
ROL TMP1 : TMP1 = ADDR<21:15>;
PRINTB #FRMT20,TMP1,TMP2 : TMP2 = ADDR<14:00>;

```

```

MOV TMP2,-(SP)
MOV TMP1,-(SP)
MOV #FRMT20,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #10,SP

```

RTS PC

... TLM UNIQUE ERROR SIGNATURES.

```

ERR20: JSR PC,E2022 : BAD STATUS AFTER READ,READNH,WRTCHK...
ERR20A: PRINTX #FRMT26,TMP1,#EM99 : ...OR AFTER SEEK,GET STATUS,WRITE.

```

```

MOV #EM99,-(SP)
MOV TMP1,-(SP)
MOV #FRMT26,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #10,SP

```

PRINTX #FRMT27 ; CRLF.

```

MOV #FRMT27,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #4,SP

```

```

JMP ALLREGS
ERR21: JSR PC,E2022 : READ HEADER FAILS.
JMP CKERLT

```

```

ERR22: JSR PC,E2022 : DATA FROM TLM WRONG.
PRINTX #FRMT14,TMP1,TMP0,GDDAT,BDDAT

```

```

MOV BDDAT,-(SP)
MOV GDDAT,-(SP)
MOV TMP0,-(SP)
MOV TMP1,-(SP)
MOV #FRMT14,-(SP)
MOV #5,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #14,SP

```

PRINTX #FRMT27 ; CRLF

```

MOV #FRMT27,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #4,SP

```

```

JMP ALLREGS
E2022: PRINTX #FRMT25,#TSEC,TMP2,#TPEAK

```

CVRLBAO -- RLV12 DICKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(063) 08-FEB-82 10:07 K 9
GLOBAL ERROR HANDLERS AND ASCII TEXT

SEQ 0114

(10) 027042 012746 036602
(9) 027046 013746 003430
(8) 027052 012746 036523
(7) 027056 012746 030401
(6) 027062 012746 000004
(3) 027066 010600
(4) 027070 104414
(4) 027072 062706 0C0012
3162 027076 000207

MOV #TPEAK, -(SP)
MOV TMP2, -(SP)
MOV #TSEC, -(SP)
MOV #FRMT25, -(SP)
MOV #4, -(SP)
MOV SP, R0
TRAP C\$PNTX
ADD #12, SP

RTS PC

...
: DISPLAY ALL REGISTERS BEFORE AND AFTER ERROR (IF REQ'D),
: CHECK IF ERROR LIMIT EXCEEDED AND EXIT ACCORDINGLY.
...

ALLREGS: CLR P0 ; DISABLE...

CMP RLTP, #RLV12X

BNE 1\$

MOV #1, R0

1\$: MOVB R0, FRMT1A ; ...OR ENABLE THE EXTRA REGISTER?

MOVB R0, FRMT1C ; ENABLE/DISABLE BAE TEXT.

PRINTB #FRMT1 ; HEADER.

MOV #FRMT1, -(SP)
MOV #1, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #4, SP

PRINTB #FRMT1B, #BEREG, B.CS, B.BA, B.DA, B.MP, B.BAE

MOV B.BAE, -(SP)
MOV B.MP, -(SP)
MOV B.DA, -(SP)
MOV B.BA, -(SP)
MOV B.CS, -(SP)
MOV #BEREG, -(SP)
MOV #FRMT1B, -(SP)
MOV #7, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #20, SP

PRINTB #FRMT1B, #AFREG, E.CS, E.BA, E.DA, E.MP, E.BAE

MOV E.BAE, -(SP)
MOV E.MP, -(SP)
MOV E.DA, -(SP)
MOV E.BA, -(SP)
MOV E.CS, -(SP)
MOV #AFREG, -(SP)
MOV #FRMT1B, -(SP)
MOV #7, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #20, SP

PRINTB #FRMT1D

MOV #FRMT1D, -(SP)
MOV #1, -(SP)
MOV SP, R0
TRAP C\$PNTB
ADD #4, SP

; FALL THRU...

```

3178      ;...AND CHECK ERROR LIMIT (IF ANY).
3179      ;KERLT: INLOOP
3180      ;
3181      027306 104420      BCOMPLETE 3$      ; RETURN IF LOOPING...      TRAP  C$INLP
3182      (3) 027306      BCS 3$
3183      (2) 027310 103425      TST  ERRLMT      ;...OR NO ERROR LIMIT SET.
3184      027312 005737 003304      BEQ  3$      ;COUNT THE UNIT ERROR DETECTED
3185      027316 001422      INC  @ERPOINT      ;REACHED THE ERROR LIMIT?
3186      027320 005277 154142      CMP  @ERPOINT,ERRLMT ;NO, RETURN
3187      027324 027737 154136 003304      BLT  3$      ; 'ERROR LIMIT EXCEEDED'
3188      027332 002414      PRINTF #FRMT23      MOV  #FRMT23, -(SP)
3189      (7) 027334 012746 030350      MOV  #1, -(SP)
3190      (6) 027340 012746 000001      MOV  SP, R0
3191      (3) 027344 010600      TRAP C$PNTF
3192      (4) 027346 104417      ADD  #4, SP
3193      (4) 027350 062706 000004      2$: DODU  UNITST      ;DROP THE UNIT...      MOV  UNITST, R0
3194      (3) 027354 013700 003322      TRAP C$DODU
3195      (3) 027360 104451      DOCLN      ;...AND ABORT.      TRAP  C$DCLN
3196      (3) 027362 104444      3$:  ENDMSG      ; OR RETURN TO CALLER.      L10063: TRAP  C$MSG
3197      (3) 027364 104423

```

3193
3194
3195
3196
3197
3198
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248

027366	040445	042522	044507
027425	045	004501	020040
027436	047045	052045	047445
027467	045	031123	047445
027476	047045	000	
027501	045	040501	042104
027517	045	042501	051122
027541	045	042501	050130
027602	040745	054105	023520
027636	040445	052101	044440
027670	040445	052101	050040
027723	045	047501	044520
027766	040445	054105	042520
030031	045	022516	033501
030120	047045	042045	022463
030201	045	040501	020124
030241	045	042501	050130
030276	040445	047527	042122
030350	040445	051105	047522
030401	045	022524	022524
030410	047045	052045	040445
030431	045	000116	
030434	042502	047506	042522
030455	101	052106	051105
030476	047503	052116	047522
030523	104	044522	042526
030551	040	051105	000122
030556	042040	053122	000
030563	040	054116	000115
030570	050040	051101	000
030575	040	046104	000124
030602	044040	043116	000
030607	040	041504	041522
030615	040	041510	041522
030623	040	050117	000111
030630	047040	047117	000105
030636	047503	052116	047522
030706	040503	020116	047516
030733	103	047101	047040
030760	040503	020116	047516
031005	103	047101	047040
031032	040503	020116	047516
031060	040502	020105	044123
031127	123	041514	020123
031200	046122	040502	051040
031226	046122	040502	051040
031254	046122	040502	020105

FORMATTED ASCII TEXT.

```

FRMT1: .ASCII /%REGISTERS      CS      BA      DA      MP/
FRMT1A: .ASCII /%A      BAE/
FRMT1B: .ASCII /%NXT%06XS2%06XS2%06XS2%06/
FRMT1C: .ASCII /%S2%06/
FRMT1D: .ASCII /%N/
FRMT2: .ASCII /%AADDR: %06XN/
FRMT3: .ASCII /%AERRORS SET: %TXN/
FRMT3A: .ASCII /%AEXP'D: ERR HNF OPI REC'D: %TXN/
FRMT4: .ASCII /%AEXP'D: %06XA REC'D: %06XN/
FRMT5: .ASCII /%AAT INTERNAL STATE %D2XN/
FRMT6: .ASCII /%AAT PROCESSOR LEVEL %01XN/
FRMT10: .ASCII /%AOPI FLAG RECEIVED AT%D4XA MSEC.%N/
        .ASCII /%AEXPECTED FROM 155 TO 650 MSEC.%N/
FRMT14: .ASCII /%N%WORD: %03XA ADDR: %06XA EXP'D: %06XA REC'D: %06/
FRMT15: .ASCII /%N%D3XA WORDS BAD OUT OF 255 WORDS TRANSFERRED%N/
FRMT20: .ASCII /%AAT PHYSICAL ADDRESS: %03%05XN/
FRMT21: .ASCII /%AEXP'D: ERR NXM REC'D: %TXN/
FRMT22: .ASCII /%AWORD: %03XA EXP'D: %06XA REC'D: %06XN/
FRMT23: .ASCII /%AERROR LIMIT EXCEEDED%N/

FRMT25: .ASCII /%TX%TX%TX/
FRMT26: .ASCII /%NXTXA REC'D: %T/
FRMT27: .ASCII /%N/
        .EVEN

```

PURE (UNFORMATTED) ASCII TEXT.

```

BEREG: .ASCII /BEFORE COMMAND: /
AFREG: .ASCII /AFTER COMMAND: /
CRTIM: .ASCII /CONTROLLER TIMED OUT/
DRTIM: .ASCII /DRIVE READY TIMED OUT/
CERR: .ASCII / ERR/
DEMES: .ASCII / DRV/
NXMMES: .ASCII / NXM/
PARMES: .ASCII / PAR/
DLTMES: .ASCII / DLT/
HNFMES: .ASCII / HNF/
DCRCMES: .ASCII / DCRC/
HCRMES: .ASCII / HCRC/
OPIMES: .ASCII / OPI/
NON: .ASCII / NONE/

EM0: .ASCII /CONTROLLER DOESN'T ANSWER -- CAN'T TEST/
EM1: .ASCII /CAN NOT ADDRESS RLCS/
EM2: .ASCII /CAN NOT ADDRESS RLBA/
EM3: .ASCII /CAN NOT ADDRESS RLDA/
EM4: .ASCII /CAN NOT ADDRESS RLMP/
EM4A: .ASCII /CAN NOT ADDRESS RLBAE/
EM4B: .ASCII /BAE SHOULD BE DISABLED -- IT'S NOT !!!/
EM5: .ASCII \RLCS READ/WRITE ERROR (BIT 0 DON'T CARE)\
EM6: .ASCII \RLBA READ/WRITE ERROR\
EM7: .ASCII \RLDA READ/WRITE ERROR\
EM8: .ASCII \RLBAE READ/WRITE ERROR\

```

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 N 9
GLOBAL ERROR HANDLERS AND ASCII TEXT

SEQ 0117

```
3249 031303 122 041114 020101 EM10: .ASCII /RLBA ERROR AFTER MAINT. FUNCTION/
3250 0031344 046123 040504 022420 EM12: .ASCII /RLDA ERROR AFTER MAINT. FUNCTION/
3251 0031405 122 041514 020123 EM14: .ASCII /RLCS INCORRECT AFTER NO-OP/
3252 0031440 047516 047455 020120 EM14A: .ASCII /NO-OP ALTERED RLBA, RLDA, OR RLMP/
3253 0031502 040515 047111 042524 EM14B: .ASCII /MAINTENANCE MODE SEQUENCER FAILED/
3254 0031544 047516 044440 052116 EM15: .ASCII /NO INTERRUPT ON FUNCTION COMPLETE/
3255 0031606 047111 042524 051122 EM16: .ASCII /INTERRUPT PRIORITY FAILURE/
3256 0031641 106 051117 042503 EM17: .ASCII /FORCED OPI -- INTERRUPT NOT RECEIVED/
3257 0031706 051103 020103 043117 EM20: .ASCII /CRC OF DA+3 INCORRECT (SERIAL DATA PATH)/
3258 0031757 103 041522 047440 EM21: .ASCII /CRC OF CRC OF DA+4 INCORRECT (SERIAL DATA PATH)/
3259 0032037 106 046111 027514 EM22: .ASCII /FILL/EMPTY FIFO DATA TRANSFER ERROR/
3260 032103 106 046111 027514 EM23: .ASCII /FILL/EMPTY FIFO LAST WORD+1 INCORRECT/
3261 032151 105 051122 051117 EM27: .ASCII /ERROR STATUS INCORRECT AFTER FORCED (01/15><12>
3262 032223 047111 052111 040511 .ASCII /INITIAL WORD COUNT WAS /
3263 0032233 074 032440 030461 EM27X: .ASCII /< 511./
3264 0032233 050117 020111 046106 EM31: .ASCII /OPI FLAG NOT RECEIVED/
3265 0032233 050117 020111 044524 EM32: .ASCII /OPI TIMING INCORRECT/
3266 0032233 127 044522 044524 EM44: .ASCII /WRITING RLMP MODIFIED RLCS/
3267 0032233 051127 052111 047111 EM45: .ASCII /WRITING RLMP MODIFIED RLBA/
3268 0032233 127 044522 044524 EM46: .ASCII /WRITING RLMP MODIFIED RLDA/
3269 0032233 051127 052111 047111 EM47: .ASCII /WRITING RLMP MODIFIED RLBAE/
3270 0032233 046122 051503 053440 EM50: .ASCII /RLCS WRONG AFTER WRITING RLBAE/
3271 0032233 122 041114 042501 EM51: .ASCII /RLBAE MODIFIED RLBA/
3272 0032233 122 041114 042501 EM52: .ASCII /RLBAE MODIFIED RLDA/
3273 0032233 042502 020124 EM61: .ASCII /'BIT SET' ON RLCS YIELDS WRONG RESULT/
3274 0032233 042502 020124 EM62: .ASCII /'BIT CLEAR' ON RLCS YIELDS WRONG RESULT/
3275 0032233 042502 020124 EM63: .ASCII /'BIT SET' ON RLBA YIELDS WRONG RESULT/
3276 0032233 042502 020124 EM64: .ASCII /'BIT CLEAR' ON RLBA YIELDS WRONG RESULT/
3277 0032233 042502 020124 EM65: .ASCII /'BIT SET' ON RLDA YIELDS WRONG RESULT/
3278 0032233 042502 020124 EM66: .ASCII /'BIT CLEAR' ON RLDA YIELDS WRONG RESULT/
3279 0032233 042502 020124 EM67: .ASCII /'BIT SET' ON RLBAE YIELDS WRONG RESULT/
3280 0032233 041103 051114 041440 EM68: .ASCII /'BIT CLEAR' ON RLBAE YIELDS WRONG RESULT/
3281 0032233 103 051033 051033 EM69: .ASCII /BUS-RESET DIDN'T CLEAR RLCS/
3282 0032233 103 051033 051033 EM70: .ASCII /BUS-RESET DIDN'T CLEAR RLBA/
3283 0032233 103 051033 051033 EM71: .ASCII /BUS-RESET DIDN'T CLEAR RLDA/
3284 0032233 051 051033 051033 EM71A: .ASCII /BUS-RESET DIDN'T CLEAR RLBAE/
3285 0032233 046 044522 044522 EM72: .ASCII /WRITING RLCS MODIFIED RLBA/
3286 0032233 046 044522 044522 EM73: .ASCII /WRITING RLCS MODIFIED RLDA/
3287 0032233 046 044522 044522 EM73A: .ASCII /RLBAE WRONG AFTER WRITING RLCS/
3288 0032233 046 044522 044522 EM74: .ASCII /WRITING RLBA MODIFIED RLCS/
3289 0032233 046 044522 044522 EM75: .ASCII /WRITING RLBA MODIFIED RLDA/
3290 0032233 046 044522 044522 EM75A: .ASCII /WRITING RLBA MODIFIED RLBAE/
3291 0032233 046 044522 044522 EM76: .ASCII /WRITING RLDA MODIFIED RLCS/
3292 0032233 051 044522 044522 EM77: .ASCII /WRITING RLDA MODIFIED RLBA/
3293 0032233 127 044522 044522 EM77A: .ASCII /WRITING RLDA MODIFIED RLBAE/
3294 0032233 116 044522 044522 EM80: .ASCII /NXM ERROR FLAG DIDN'T SET/
3295 0032233 0200 044522 044522 EM80X: .ASCII / BANK SELECT 7 (BBS7) FAILS/
3296 0032233 040503 052116 052116 EM90: .ASCII /STATUS INCORRECT AFTER NON-EX MEMORY ACCESS/
3297 0032233 040503 052116 052116 EM91: .ASCII /UNEXPECTED STATUS ERROR ON EXT MEMORY ACCESS/
3298 0032233 040503 052116 052116 EM92: .ASCII /DATA INCORRECT TO/FROM EXTENDED MEMORY/
3299 0032233 040503 052116 052116 EM98: .ASCII /UNEXPECTED CONTROLLER ERRORS/
3300 034404 000170 EM99: .BLKB 120.
3301
3302
3303
3304 034574 052502 020123 044524 EM100: .ASCII /BUS TIME-OUT ON TLM ADDRESS/
```

CVRLBA0 -- RLV12 DISKLESS
CVPLBA.P:1 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 B 10
GLOBAL ERROR HANDLERS AND ASCII TEXT

SEQ 0118

```
3305 034630 046124 020115 052123 EM100A: .ASCII /TLM STATUS WRONG/<15><12>
3306 034630 046124 047103 020115 .ASCII /CHECK THAT CABLE IS IN PLACE !!!/
3307 034630 046124 047103 020115 EM100B: .ASCII /PROM ID WRONG/<15><12>
3308 034630 046124 047103 020115 .ASCII /CHECK THAT CORRECT PROM SET IS INSTALLED !!!/
3309 034630 046124 047103 020115 EM101: .ASCII \SYSClk<11> AND/OR PWRCK<8> NOT SET IN TLM STATUS\
3310 034630 046124 047103 020115 EM102: .ASCII /DRIVE READY STATUS BIT<0> INCORRECT IN RLCSR/
3311 034630 046124 047103 020115 EM103: .ASCII /DRIVE ERROR STATUS BITS<15:14> INCORRECT IN RLCSR/
3312 034630 046124 047103 020115 EM104: .ASCII /DRIVE /
3313 034630 046124 047103 020115 EM104X: .ASCII /0 SELECT BITS<10:9> INCORRECT IN TLM STATUS/
3314 034630 046124 047103 020115 EM105: .ASCII /DRIVE COMMAND (GET STATUS) INCORRECT IN TLM CSR+2/
3315 034630 046124 047103 020115 EM106: .ASCII /RETURNED DRIVE STATUS INCORRECT IN RLMP/
3316 034630 046124 047103 020115 EM107: .ASCII /ERRORS IN RLCSR AFTER 'GET STATUS' COMMAND/
3317 034630 046124 047103 020115 EM110: .ASCII /DRIVE COMMAND (SEEK DIFF) INCORRECT IN TLM CSR+2/
3318 034630 046124 047103 020115 EM111: .ASCII /ERRORS IN RLCSR AFTER 'SEEK' COMMAND/
3319 034630 046124 047103 020115 EM112: .ASCII /WRITE GATE<13> OR DATA ACTIVE<12> DIDN'T SET DURING WRITE/
3320 034630 046124 047103 020115 EM113: .ASCII /WRITE GATE ERROR BIT<14> SET DURING WRITE/
3321 034630 046124 047103 020115 EM113A: .ASCII /WRITE GATE BIT<13> STILL SET AFTER WRITE DONE/
3322 034630 046124 047103 020115 EM114: .ASCII /ERRORS IN RLCSR AFTER WRITE DATA COMMAND/
3323 034630 046124 047103 020115 EM115: .ASCII /ERROR STATUS INCORRECT AFTER READ DATA/
3324 034630 046124 047103 020115 EM115X: .ASCII <0>/WITHOUT HEADER CHECK/
3325 034630 046124 047103 020115 EM116: .ASCII /READ HEADER CAN'T FIND TLM SECTOR 2 (CYL1,SEC6)/
3326 034630 046124 047103 020115 EM117: .ASCII /RECEIVED DATA INCORRECT AFTER READ/
3327 034630 046124 047103 020115 EM117X: .ASCII <0>/WITHOUT HEADER CHECK/
3328 034630 046124 047103 020115 EM118: .ASCII /ERROR STATUS INCORRECT AFTER WRITE-CHECK/
3329 034630 046124 047103 020115 EM119: .ASCII /PROM READ FAILURE/<15><12>
3330 034630 046124 047103 020115 .ASCII /CAN'T FIND SECTOR /
3331 034630 046124 047103 020115 EM119X: .ASCII /1/
3332 034630 046124 047103 020115 TSEC: .ASCII /SECTOR /
3333 034630 046124 047103 020115 TSECX: .ASCII /1, /
3334 034630 046124 047103 020115 TCLK0: .ASCII /NOMINAL CLOCK/
3335 034630 046124 047103 020115 TCLK1: .ASCII /FAST CLOCK/
3336 034630 046124 047103 020115 TCLK2: .ASCII /SLOW CLOCK/
3337 034630 046124 047103 020115 TPEAK: .ASCII / - MAX PEAK SHIFT/
3338 034630 046124 047103 020115 EXPHCRC: .ASCII /EXP'D: ERR HCRC OPI/
3339 034630 046124 047103 020115 EXPDCRC: .ASCII /EXP'D: ERR DCRC/
3340 034630 046124 047103 020115 EXPNON: .ASCII /EXP'D: NONE/
3341 034630 046124 047103 020115 .EVEN
```



```
3343 .SBTTL DEVELOPMENT/DEBUG AIDS
3344 .SBTTL
3345 .SBTTL RLV12 EMULATOR
3346
3347 : FOR DEBUGGING (UNDER MIMIC) WE'LL EMULATE THE ACTION
3348 : THAT WOULD OCCUR ON A MAINT/NOP FUNCTION.
3349
3350 036704 036704
3351 036704 000000
3352
3353 .PRINT 0 ; FOR DEBUG, MIMIC FLAG IS HERE...
3354 MIMIC: 0 ;...SET NZ TO ENABLE EMULATOR.
3355
3356 036706 023727 003402 000000 EMURLV: CMP RLTP,#RL11
3357 036714 001004 BNE 1$ ; BR IF RLV
3358 036716 042777 036000 144400 BIC #36000,@RLCS ; RL11 -- NOP -- CLEAR ERROR BITS.
3359 036724 000517 BR 7$ ; AND THAT'S ALL.
3360
3361 036726 027727 144400 177001 1$: CMP @RLMP,#-511. ; WORD COUNT RIGHT ??
3362 036734 001404 BEQ 2$ ; YES.
3363 036736 052777 112000 144360 BIS #ERR!MNF!OPI,@RLCS ; NO SET THE ERROR BITS...
3364 036744 000507 BR 7$ ;...AND RETURN.
3365
3366 036746 020527 020416 2$: CMP R5,#BB7 ; CALLED FROM BBS7 TEST ??
3367 036752 001004 BNE 3$ ; SKIP IF NOT.
3368 036754 052777 120000 144342 BIS #ERR!NXM,@RLCS ; YES, SET ERROR BITS...
3369 036762 000500 BR 7$ ;...AND RETURN.
3370
3371 036764 012701 004364 3$: MOV #BUF1,R1 ; STANDARD SRC...
3372 036770 012702 005364 MOV #BUF2,R2 ; ...AND DST...
3373 036774 012703 037236 MOV #DFIFO,R3 ; ...AND DUMMY FIFO POINTERS.
3374 037000 023701 010352 CMP BA16,R1 ; IS BA STANDARD (INTERNAL) ??
3375 037004 001411 BEQ 4$ ; BR IF SO.
3376 037006 012701 100000 MOV #100000,R1 ; POINT TO BUF1 IF EXT MEMORY.
3377 037012 012702 101000 MOV #101000,R2 ; DITTO BUF2 (RECEIVER).
3378 037016 012737 037106 000004 MOV #5$ ERRVEC ; SET TRAP CATCHER.
3379 037024 005237 177572 INC MMRO ; ***** KT ON *****
3380 037030 062777 000002 144270 4$: ADD #2,@RLBA ; SIMULATE INC BA...
3381 037036 005577 144272 ADC @RLBAE ; ...OVERFLOWING INTO BAE.
3382 037042 012123 MOV (R1)+,(R3)+ ; MEM => FIFO (256 WORDS).
3383 037044 060240 240
3384 037046 020527 040236 CMP R3,#DFIFO+1000 ;
3385 037052 103766 4$ ;
3386 037054 012703 37236 BLO 4$ ;
3387 037056 062777 000002 144240 14$: MOV #DFIFO,R3 ; RESET FIFO POINTER.
3388 037060 005577 144242 ADD #2,@RLBA ;
3389 037066 012322 MOV @RLBAE ; INC BA.
3390 037072 000240 240 (R3)+,(R2)+ ; FIFO => MEM (255 WORDS).
3391 037074 020527 040234 CMP R3,#DFIFO+776 ;
3392 037076 103766 14$ ;
3393 037078 000406 6$ BLO 6$ ; LOOP...
3394 037080 052777 120000 144210 5$: BIS #ERR!NXM,@RLCS ; ...UNTIL DONE.
3395 037086 012716 037122 MOV #6$,(SP) ; ON NXM TRAP, SET ERROR BITS...
3396 037092 000002 RTI ; ...AND CONTINUE.
3397 037094 005037 177572 6$: CLR MMRO ; ***** KT OFF *****
3398 037096 012737 010624 000004 MOV #TRAP4,ERRVEC ; RESET TRAP CATCHER.
3399 037102 017700 144170 MOV @RLDA,R0 ;
3400 037104 062700 000006 ADD #6,R0 ; ADJUST FINAL DA...
3401 037106 110077 144160 MOVB R0,@RLDA ; ...LO BYTE ONLY.
```

CVRLBAO -- RLV12 DISKLESS MACY11 30G(1063) 08-FEB-82 10:07 D 10
CVRLBA.P11 08-FEB-82 09:59 RLV12 EMULATOR PAGE 47-1

SEQ 0120

3399	037150	013777	003442	144154		MOV	GDCRC3, @RLMP	:	FAKE THE 1ST...
3400	037156	013737	003444	003374		MOV	GDCRC4, E.MP1	:	...AND 2ND CRC WORDS.
3401	037164	052777	00020C	144132	7\$:	BIS	#CRDY, @RLCS	:	SET DONE BIT.
3402	037172	032777	000100	144124		BIT	#INTEN, @RLCS	:	INTERRUPT EXPECTED ??
3403	037200	001410				BEQ	8\$	:	NO.
3404	037204	04440				GETPRI	RO		
3405	037204	42700	177437			BIC	#^CPRI07, RO	:	MASK PRIORITY BITS. TRAP CSGPRI
3406	037204	J20037	003340			CMP	RO, BFPRI0		
3407	037214	002002				BGE	8\$	:	BR IF CPU >= RL BR LEVEL
3408	037216	005237	010634			INC	INTFLG	:	OTHERWISE, SET INT RECEIVED FLAG.
3409	037222	000207			8\$:	RTS	PC	:	...AND RETURN.
3410									
3411	037224	000005			DUMMY:	.BLKW	5	:	THESE ARE THE DUMMY REGISTERS.
3412	037236	000400			DFIFO:	.BLKW	256.	:	AND A DUMMY FIFO.

MACY11 30G(1063) 08-FEB-82 10:07 E 10
RLV12 EMULATOR PAGE 47-2

3414		
3415		
3416		
3417	040236	040236
3418		040340
3419		
3420		
3421		
3422	040340	
(2)		
(2)	040340	040362
(2)	040342	000007
(2)	040344	
3423		
3424		
3425		
3426		
3427		
3428		
3429	040344	
3430	040344	
(4)	040344	000000
(3)	040346	000005
(3)	040350	
3431	040350	174400
3432	040352	000160
3433	040354	000200
3434	040356	000000
3435	040360	000003
3436	040362	
(3)	040362	
3437	040362	
3438		
3439		
3440		000001

PATCH: := .+100

LASTAD ; LAST USED ADDRESS.

```

.EVEN
.WORD TSFREE
.WORD TSSIZE

```

LSLAST::

;

```

; NOW CAN A SINGLE UNIT P-TABLE, SO WE CAN RUN UNDER
; DEFAULT CONDITIONS, AND CALL THE BINARY A .BIC FILE.

```

BGNSETUP 1
BGNPTAB

```

: 1 UNIT ONLY...
:...USING THE FOLLOWING DEFAULTS.

```

.WORD 0
.WORD L10066-./2-1

L10064:

```

.WORD 174400
.WORD 160
.WORD PR104
.WORD 0
.WORD 3

```

```

: (0) CSR ADDRESS.
: (2) VECTOR.
: (4) PRIORITY.
: (6) DRIVE (BITS 8,9,10).
: (10) CONTROLLER TYPE (RLV12 W/BAE).

```

L10066:

ENDSETUP

.....

.END

SEQ 0122

2224	2224	2224	2224	2224	2224	2224	2224	2224	2224	2224	2224	2224
709#	785	436	2472	2507	2612	2638	2729	3104	3107	3110	3122	3133
278#	1322	1348	1374	1442	1478	1514						3157
317#	3223#											
58	60	100	103	110	147#	181	185	1327	2570	2695		
147#												
147#												
147#												
147#												
147#												
147#												
147#												
147#												
147#												
56	109	147#	174	176	178	180	184	350	2565	2690		
147#	165	2408										
147#	163	164										
147#	161	162	2512									
147#	158	160	2512	2523	2526							
107	147#	158	2523	2575	2700							
106	147#	157	3140									
108	147#	175	176	179	180	186	351	2560	2685			
147#	177	178	179	180	183	352						
147#	189	353										
147#	354											
147#	171	187	188	355								
147#	170											
147#	167	169	356									
147#	168	169	357	2408								
230#	529*	568	1838	3406								
63	75#											
450#	2030	2983	2986	3003								
448#	554	792	808	832	2059	2109	2122	2155	2226	2266	2351	2505

SEQ 0123

BUF2	005364	2646	2728	2901	2929	2943	2962	3001	3035	3368						
BVEC	003342	231#	2060	2135	2156	2602	2611	2627	2984	3005	3369					
B.BA	003352	235#	211*	1779	3174											
B.BAE	003360	238#	216*	3174												
B.CS	003370	237#	209*	1772	3174											
B.DA	003374	236#	212*	1781	2036	3174	3174									
B.MP	003356	237#	213*	1771*	1783	3174										
CALBCC	003410	250#	926	934	939	980*										
CALCRC	010714	616#	1761	1813	1842	1876	1909	1939	1984	1999	2011	2205	2281			
CANTFI	025731	2959#	3063#													
CERR	030551	653#	3227#													
CHKDAT	023274	2622#	2642#	2646#												
CKERLT	027306	3097#	3102#	3105	3117	3123	3154	3181#								
CLKRTI	010636	571#	885#													
CLRTLH	024132	2431#	2467#	2498	2762	2782#										
CONT	007050	476#	552#													
CPUTYP	003400	246#	504*	541	1937											
CRCEND	004270	412#														
CRDY =	000200	170#	594	599	769	786	893	1251	1401	1403	1558	1590	1631	1672		
		1721#	2404	2419	2434	2437	2470	2473	2502	2508	2608	2613	2627	2628		
		2636#	2639	2725	2730	3401										
CRLF	025760	3065#														
CRTIM	030476	900#	3225#													
CSEND	004176	380#	1308	1428												
CSPAT	004100	349#	1293	1399												
CSRA	002323	61#	73#													
CSAU =	000052	17#	628													
CSAUTO=	000061	17#	588													
CSBRK =	000022	17#														
CSBSEG=	000004	17#	1294	1323	1349	1375	1400	1443	1479	1515	1839	2019	2133	2265		
		2601#	2724													
CSBSUB=	000002	17#	1978	1993	2008	2106	2119									
CSCEFG=	000045	17#														
CSCLCK=	000062	17#														
CSCLEA=	000012	17#	604													
CSCLOS=	000035	17#														
CSCLP1=	000006	17#	1564	1570	1605	1611	1646	1652	1688	1694	1700	1735	1740	1778		
		1786#	1797	1855	1891	1919	2026	2035	2045	2051	2057	2073	2147	2153		
		2169#	2340													
CSCVEC=	000036	17#														
CSDCLN=	000044	17#	513	1137	1159	1181	1203	1228	3049	3190						
CSDODU=	000051	17#	587	1136	1158	1180	1202	1227	3189							
CSDRPT=	000024	17#	509													
CSDU =	000053	17#	616													
CSEDIT=	000003	17#	42													
CSERDF=	000055	17#	905	1232	1269	1273	1276	1281	1305	1332	1355	1382	1412	1425		
		1453#	1462	1486	1495	1523	1533	1563	1569	1577	1604	1610	1618	1645		
		1651#	1659	1687	1695	1699	1707	1734	1739	1743	1777	1785	1796	1821		
		1854#	1890	1918	1949	1956	2025	2034	2044	2050	2056	2070	2086	2146		
		2152#	2166	2224	2305	2310	2328	2376	2383	2391	2399	2415	2447	2452		
		2456#	2483	2587	2518	2528	2530	2534	2558	2591	2634	2644	2655	2683		
		2737#	2769	2771	2773											
		17#														
CSEHHR=	000056	17#														
CSEHRO=	000060	17#														
CSEHRSF=	000054	17#	586	1135	1157	1179	1201	1226								

MACV11 306(1063) 08-FEB-82 10:07 H 10 PAGE 48-2
CROSS REFERENCE TABLE -- USER SYMBOLS

H 10
PAGE 48-2

17#	1306	1333	1356	1383	1413	1426	1454	1463	1487	1496	1524	1534
17#	1310	1337	1360	1387	1413	1426	1454	1463	1487	1496	1524	1534
2662	2738	2758	2788	2815	2843	2867	2890	2918	2940	2968	2991	3014
17#	1989	2004	2014	2115	2128	2144	2168	2183	2207	2228	2251	2274
17#	1138	1160	1182	1204	1233	1282	1311	1318	1361	1388	1431	1468
1501	1539	1578	1619	1660	1708	1744	1798	1823	1861	1898	1921	1957
2090	2182	2228	2353	2420	2457	2488	2535	2664	2740			
17#	1787	2015	2029	2129	2201	2253	2595	2720				

2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431	2432	2433	2434	2435	2436	2437	2438	2439	2440	2441	2442	2443	2444	2445	2446	2447	2448	2449	2450	2451	2452	2453	2454	2455	2456	2457	2458
------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------

[illegible][illegible]

1834#	1873#	1873#	1722#	1724#	2395#								
3370#	3381#	3383#	3388#	3412#									
197#	900#	1232#	1269#	1273#	1276#	1281#	1305#	1332#	1355#	1382#	1412#	1425#	

2853 [#]	2885 [#]	2938 [#]	2989 [#]
17			
161 [#]	670		
672	677	3231 [#]	

MACY11 30G(1063) 08-FEB-82 10:07 1 10
CROSS REFERENCE TABLE -- USER SYMBOLS

I 10
PAGE 48-3

Variable	Value	Variable	Value	Variable	Value	Variable	Value	Variable	Value	Variable	Value	Variable	Value	Variable	Value	Variable	Value	Variable	Value
DLVCM	003462	271#	549*	746															
DMA1	016762	1678#																	
DMA2	017030	1993#																	
DMA3	017076	2008#																	
DMODE	024566	2884#	2945*	2948*	2954	2967													
DOIT	025064	2950#																	
DRDY =	000001	181#	1266	1302	1409	1422	1601	1642	1684	1731	2379	23*							
DRIVE	003346	233#	530*	768															
DRN	002363	65#	76#																
DROPPE	003464	272#	485*																
DRST =	000010	183#	2435	511	609	611*	621	623*											
DRTIM	030523	2226#	2444																
DSO	000000	166#																	
DS1 =	000400	167#	2416																
DS2 =	001000	168#																	
DS3 =	001400	169#																	
DTABL	024572	2853	2886#																
DUMMY	037224	527	3411#																
EF.CON=	000036	147#	473																
EF.NEW=	000035	147#	471																
EF.PWR=	000034	147#	465																
EF.RES=	000037	147#	469																
EF.STA=	000040	147#	467																
EMJRLV	036706	789	3353#																
EMO	030636	586	3238#																
EM1	030706	1135	3239#																
EM10	031303	2034	3249#																
EM100	034574	2769	3304#																
EM100A	034630	2771	3305#																
EM10GB	034713	2773	3307#																
EM101	035007	2376	3309#																

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 05:59

MACY11 30G(1063) 08-FEB-82 10:07 J 10
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0126

EM15	031544	1821	2152	3254#			
EM16	031606	1854	2255#				
EM17	031641	1918	2256#				
EM2	030733	1157	2240#				
EM20	031706	2050	2257#				
EM21	031757	2056	2258#				
EM22	032037	2070	2166	3259#			
EM23	032103	2086	2260#				
EM27	032151	1890	2261#				
EM27X	032251	1879*	2280*	1895*	1896*	3263#	
EM3	030760	1179	2241#				
EM31	032260	1949	2264#				
EM32	032306	1956	2265#				
EM4	031005	1201	2242#				
EM4A	031032	1226	2243#				
EM4B	031060	1232	2244#				
EM44	032333	1687	2266#				
EM45	032366	1693	2267#				
EM46	032421	1699	2268#				
EM47	032454	1707	2269#				
EM5	031127	1305	2245#				
EM50	032510	1734	2270#				
EM51	032547	1739	2271#				
EM52	032573	1743	2272#				
EM6	031200	1332	2246#				
EM61	032617	1412	2273#				
EM62	032665	1425	2274#				
EM63	032735	1433	2275#				
EM64	033005	1462	2276#				
EM65	033053	1486	2277#				
EM66	033121	1495	2278#				
EM67	033171	1523	2279#				
EM68	033240	1533	2280#				
EM69	033311	1269	2281#				
EM7	031226	1355	2247#				
EM70	033345	1273	2282#				
EM71	033401	1276	2283#				
EM71A	033435	1281	2284#				
EM72	033472	1563	2285#				
EM73	033525	1569	2286#				
EM73A	033560	1577	2287#				
EM74	033617	1604	2288#				
EM75	033651	1610	2289#				
EM75A	033703	1618	2290#				
EM76	033737	1645	2291#				
EM77	033772	1651	2292#				
EM77A	034025	1659	2293#				
EM8	031254	1382	2248#				
EM80	034061	2224	2294#				
EM80X	034112	2202*	2204*	3295#			
EM90	034147	2303	2296#				
EM91	034223	2310	2297#				
EM92	034300	2338	2298#				
EM98	034347	2025	2146	3299#			
EM99	034404	643	3113	3119	3126	3149	3300#
ENDDAT	004362	444#					

SEQ 0127

ENDPAT	004076		345#	1335	1358	1385	1465	1498	1536							
ENVIRN	006524		491#													
ERCOUN	003470		274#	486	514											
ERPOIN	003466		273#	514*	519*	3185*	3186									
ERR =	100000		157#	1254	1300	1407	1420	1593	1634	1675	1724	1885	2216	2300	2395	
			2827#	2833	3360	3365	3391									
ERRLMT	003304		136#	3183	3186											
ERRSIG	026160	G	3096#													
ERRVEC=	000004		151#	569*	570*	2269*	2280*	2916*	3375*	3395*						
ERR0	026164		900	1785	1918	1949	2070	2152	2166	3099#						
ERR1	026170		586	1135	1157	1179	1201	1226	1232	2769	3101#					
ERR10	026420		1890	3119#												
ERR11	026450		1956	3122#												
ERR12	026500		2224	2305	3125#											
ERR13	026534		2310	3129#												
ERR14	026544		2338	3132#												
ERR2	026220		1269	1273	1276	1281	1305	1332	1355	1382	1412	1425	1453	1462	1486	
			1495	1523	1533	1563	1569	1577	1604	1610	1618	1645	1651	1659	1687	
			1693	1699	1707	1734	1739	1743	1777	2376	2383	2391	2399	2415	2415	2771
			2773	3104#												
ERR20	026662		2620	2644	2737	3147#										
ERR20A	026666		2456	2487	2534	3148#										
ERR21	026742		2634	3153#												
ERR22	026752		2655	3156#												
ERR3	026254		1796	3107#												
ERR4	026304		2034	2044	2050	2056	2086	2447	2452	2483	2518	2528	2530	3110#		
ERR6	026340		2025	2146	3113#	3130										
ERR7	026370		1821	1854	3116#											
EVL =	000004	G	147#													
EXECUT	017132		1986	2001	2013	2019#										
EXMDUN	021262		2348	2351#												
EXMLUP	020610		2264#	2349												
EXTST	020526		2252	2254#												
EXPDCR	036650		2828	3339#												
EXPICR	036624		2621	2834	3338#											
EXPNON	036670		2455	2486	2533	2643	2852	2858	3340#							
ESEND =	002100		17#													
ESLOAD=	000035		17#	42												
E.BA	003366		241#	723*	1779	2031	2297	3175								
E.BAE	003376		245#	731*	2295	3175										
E.CS	003364		240#	644	650	654	659	662	665	670	673	678	681	686	722	
			1774	1794	1886	2027	2148	2217	2288	2301	2615	2732	3175			
E.DA	003370		242#	724*	1781	1791	2037	3175								
E.MP	003372		243#	725*	1783	2047	2448	3175								
E.MP1	003374		244#	728*	2053	3400*										
E2022	027042		3147	3153	3156	3161#										
FIFO1	017662		2106#													
FIFO2	017714		2119#													
FINDSE	025174		2854	2956	2979#											
FIX	010004		646	652	657	668	676	684	688	700#						
FRMT1	027366		3173	3196#												
FRMT1A	027425		3171*	3197#												
FRMT1B	027436		3174	3175	3198#											
FRMT1C	027467		3172*	3199#												
FRMT1D	027476		3176	3200#												
FRMT10	027723		3122	3207#												

MACY11 306(1063) 08-FEB-82 10:07 L 10 PAGE 48-6
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

M 10

PAGE 48-7
SER SYMBOLS

FSTEST= 000001

GDCRC3	003442
GDCRC4	003444
GDDAT	003432

```
GETERR      007510
GSBIT  =    000002
GSTAT  =    000004
G$CNT0 =    000200
G$DELM =    000372
G$DISP =    000003
G$EXCP =    000400
G$HILI =    000002
G$LOLI =    000001
G$NO    =    000000
G$OFFS =    000400
```

GSOF SI= 000376

```

GSPRMA= 000001
GSPRMD= 000002
GSPRML= 000000
GSRADA= 000140
GSRADB= 000000
GSRADD= 000040
GSRADL= 000120
GSRADO= 000020
GSXFER= 000004
GSYES = 000010

```

```
HCRC      = 004000
HCRCME    030615
HEADR     025670
HEXMOD    025350
HEXTXT    025763
HINUM     003452
HNF       = 010000
HNFMS     030602
HOE       = 100000
HSECT     025714
HUNIT     025701
```

G

MACY11 30G(1063) 08-FEB-82 10:07 PAGE 48-7
CROSS REFERENCE TABLE -- USER SYMBOLS

17#	1126	1138	1148	1160	1170	1182	1192	1204	1212	1233	1248	1282
1292	1311	1321	1338	1347	1361	1371	1388	1398	1431	1441	1468	1477
1501	1511	1539	1552	1578	1589	1619	1630	1660	1671	1708	1718	1744
1755	1798	1810	1825	1835	1861	1871	1898	1908	1921	1936	1957	1974
2090	2102	2182	2196	2228	2247	2353	2367	2420	2428	2457	2464	2488
2495	2535	2546	2664	2671	2740							
263#	926*	2044	3399									
264#	939*	2052	3400									
259#	585*	1134*	1156*	1178*	1200*	1216*	1251*	1254*	1267	1270*	1295*	1296*
1297	1300*	1303	1324*	1327*	1328	1330	1350*	1351	1353	1376*	1377	1379*
1380	1402*	1403*	1407*	1410	1415*	1416*	1420*	1423	1445*	1448*	1451	1456*
1457*	1460	1481*	1484	1489*	1490*	1493	1517*	1518*	1521	1526*	1527*	1528*
1531	1559*	1561	1565*	1567	1573*	1575	1590*	1593*	1594	1602	1606*	1608
1614*	1616	1631*	1634*	1635	1643	1647*	1648	1655*	1657	1672*	1675*	1676
1685	1689*	1691	1695*	1697	1703*	1705	1721*	1724*	1725	1729*	1732	1736*
1772*	1773*	1775	1838*	1847	2030*	2032	2036*	2041*	2042	2046*	2048	2052*
2054	2062*	2064	2072	2082*	2084	2158*	2160	2168	2318*	2320	2373*	2374
2380*	2381	2388*	2389	2396*	2397	2412*	2413	2444*	2445	2449*	2450	2480*
481	2513*	2514	2522*	2523*	2524	2651*	2750*	2757*	2764*	2765	3101	3104
3110	3133	3157										
641#	1888	2023	2144	2220	2303	2308	2453	2484	2531	2617	2641	2734
184#	2435	2444										
175#	2434											
17#												
17#												
17#	99											
17#												
17#												
17#												
17#												
17#	56	58	60	61	62	63	65	98	100	102	103	105
106	107	108	109	110	111							
17#	56	58	60	61	62	63	65	98	100	102	103	105
106	107	108	109	110	111							
17#	61	62	102									
17#	63	65	98	105	111							
17#	56	58	60	100	103	106	107	108	109	110		
17#												
17#	98											
17#	56	58	60	100	103	106	107	108	109	110		
17#	61	62	63	65	102	105	111					
17#	57	59	64	101	104							
17#	56	58	60	61	62	63	65	98	100	102	103	105
106	107	108	109	110								

SEQ 0130

[illegible]

CVRLBA0 -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 B 11
CR-SS REFERENCE TABLE -- USER SYMBOLS PAGE 48-9

SEQ 0131

LOT	= 000010	G	147#				
LSI	006374		49#	504#			
LSACP	002110	G	42#				
LSAPT	002036	G	42#				
LSAU	007433	G	42#	620#			
LSAUT	002070	G	42#				
LSAUTO	007214	G	42#	577#			
LSCCP	002106	G	42#				
LSCLEA	007272	G	42#	592#			
LSCO	002032	G	42#				
LSDEPO	002011	G	42#				
LSDESC	002122	G	42#	44#			
LSDESP	002076	G	42#				
LSDEVP	002060	G	42#				
LSDISP	026052	G	42#	3087#	3090#		
LSDL	002116	G	42#				
LSDTP	002043	G	42#				
LSDTYP	002034	G	42#				
LSDU	007350	G	42#	608#			
LSDUT	002072	G	42#				
LSDVTY	002142	G	42#	46#			
LSF	002052	G	42#				
LSENVI	002044	G	42#				
LSF	002102	G	42#				
LSXP1	002046	G	42#				
LSXP4	002064	G	42#				
LSXP5	002066	G	42#				
LSHARD	002172	G	42#	55#			
LSHINE	002128	G	42#				
LSHPCP	002016	G	42#				
LSHPTP	002022	G	42#				
LSHW	002376	G	42#	81#			
LSICP	002104	G	42#				
LSINIT	006374	G	42#	464#			
LSLADP	002026	G	42#				
LSLAST	040344	G	37#	42	3422#	3437	
LSLOAD	002100	G	42#				
LSLUN	002074	G	42#				
LSMREV	002050	G	42#				
LSNAME	002000	G	42#				
LSPRIO	002042	G	42#				
LSPROT	006366	G	42#	456#			
LSPT	002112	G	42#				
LSREPP	002062	G	42#				
LSREV	002010	G	42#				
LSRPT	011644	G	42#	1099#			
LSOFT	002412	G	42#	97#			
LSSPC	002056	G	42#				
LSSPCP	002020	G	42#				
LSSTP	002024	G	42#				
LSSTA	002030	G	42#				
LSW	003304	G	42#	135#			
LSTEST	002114	G	42#				
LSIML	002014	G	42#				
LSUNIT	002012	G	42#	511	520	578	609
L10000	002266		55	66#			

L10001	002410	81	91#						
L10002	002542	97	112#						
L10003	003322	135	143#						
L10004	007212	573#							
L10005	007270	588#							
L10006	007346	604#							
L10007	007430	616#							
L10008	007506	628#							
L10009	012112	1107	1113#						
L10010	012176	1138#							
L10011	012262	1160#							
L10012	012346	1182#							
L10013	012432	1204#							
L10014	012516	1226#							
L10015	013016	1248#							
L10016	013136	1270#							
L10017	013222	1292#							
L10018	013306	1314#							
L10019	013390	1336#							
L10020	013474	1358#							
L10021	013558	1380#							
L10022	014042	1402#							
L10023	014126	1424#							
L10024	014210	1446#							
L10025	014294	1468#							
L10026	014378	1490#							
L10027	014462	1512#							
L10028	014546	1534#							
L10029	014630	1556#							
L10030	014714	1578#							
L10031	014798	1600#							
L10032	014882	1622#							
L10033	014966	1644#							
L10034	015050	1666#							
L10035	015134	1688#							
L10036	015218	1710#							
L10037	015302	1732#							
L10038	015386	1754#							
L10039	015470	1776#	1798#						
L10040	015554	1798#							
L10041	015638	1820#							
L10042	015722	1842#							
L10043	015806	1864#							
L10044	015890	1886#							
L10045	015974	1908#							
L10046	016058	1930#							
L10047	016142	1952#							
L10048	016226	1974#							
L10049	016310	1996#							
L10050	016394	2018#	2090#						
L10051	016478	2040#							
L10052	016562	2062#							
L10053	016646	2084#							
L10054	016730	2106#							
L10055	016814	2128#							
L10056	016898	2150#							
L10057	016982	2172#							
L10058	017066	2194#							
L10059	017150	2216#							
L10060	017234	2238#							
L10061	017318	2260#							
L10062	017402	2282#							
L10063	017486	2304#							
L10064	017570	2326#							
L10065	017654	2348#							
L10066	017738	2370#							
MAINT =	000000	173#	1882	1914	1943	2021	2142	2214	2284
MFPT =	000007	499#	500						
MIMIC =	036704	525	726	787	3351#				
PK =	000001	185#	2435	2444	2471	2480			

MMR0	=	177572	1006#	1026	1058*	1079*	2249	2271*	2279*	2317*	2324*	3376*	3394*
MMR1	=	177574	1007#										
MMR2	=	177576	1008#										
MMR3	=	172516	1009#	1055	1056*	1063*	2259*						
MPXCLK		003312	139#	2560	2565	2570	2575	2579	2685	2690	2695	2700	2704
MSGNOT		012104	1100*	1103*	1104	1111#							
MSG2		011764	1104	1108#									
MSG3		012014	1105	1109#									
MSG4		012047	1106	1110#									
NEWPAS		006620	472	511#	521								
NLSI		006570	498	503#									
NOANS		025530	2921*	2923	3055#								
NON		030630	647	3236#									
NOOP	=	000000	172#										
NOSIG		025160	2558	2683	3097#								
NXM	=	050000	159#	662	2027	2148	2216	2300	3365	3391			
NXMMES		030563	664	669	3229#								
NXTU		006662	475	518#	523								
OCTMOD		025324	2884	2945	3012#								
OCTTXT		025770	3012	3067#									
OPI	=	002000	165#	645	673	681	686	1794	1885	1947	2027	2148	2833 3360
OPIMES		030623	689	3235#									
OPIMN		003446	265#	1953									
OPIMX		003450	266#	1951									
OSAPTS	=	000000	17#	42									
OSAU	=	000001	17#	41#	42								
OSBGR	=	000001	17#	41#	42								
OSBGNS	=	000001	17#	41#	42								
OSDU	=	000001	17#	41#	42								
OSERRT	=	000000	17#	42									
OSGNSW	=	000001	17#	41#	42								
OSPOIN	=	000001	17#	41#	42								
OSSETU	=	000001	17#	41#	42	3422							
PAR	=	020000	160#	662									
PARMES		030570	667	3230#									
PAR4	=	172550	2246#	2263*	2329	2344*	2345						
PATCH		040236	3417#										
PATCRC		004200	384#	1979	1994								
PATDAT		004272	416#										
PDSW		003306	137#	480	2893*								
PNT	=	001000	147#										
PRGSIZ	=	000100	37#										
PRI	=	002000	147#										
PRI00	=	000000	147#	1812	1911	2140							
PRI01	=	000040	147#										
PRI02	=	000100	147#										
PRI03	=	000140	147#										
PRI04	=	000200	84	147#	3433								
PRI05	=	000240	147#										
PRI06	=	000300	147#										
PRI07	=	000340	147#	570	572	603	1249	1822	1836	1860	1920	2179	3405
PRM		003047	105	124#									
PRMDMP		003024	103	123#									
PROMID		003310	138#	2764									
PRTPA		026610	3125	3129	3132	3138#							
PSERR		024506	2615	2732	2827*	2833*	2851*	2859#					

SEQ 0134

PSETXT	024504	2619	2621	2736	2828*	2834*	2852*	2858#										
PSX	024214	2554	2679	2806#														
PS1	024252	2811	2821#															
PS2	024270	2812	2825#															
PS3	024322	2813	2831#															
PS4	024332	2814	2837#															
PS5	024372	2815	2841#															
PS6	024442	2816	2849#															
PS7	024410	2817	2845#															
PURE	025774	3051	3068#															
Q0	025561	3050	3056#															
Q1	025612	3027	3058#															
Q2	025637	2941	3059#															
RAND	010526	834	854#															
RDHDR =	000010	177#	2627															
RDNHDR=	000016	180#	2636															
READ =	000014	179#	2608															
READDU	023356	2635	2662#															
READNH	023130	2624#																
READSE	022766	2578	2583	2601#														
RESTAR	006452	466	470	478#														
RHHS =	000100	187#																
RLBA	003326	225#	559*	711	723	771*	1150*	1152	1156	1255*	1271	1328*	1329	1444*				
		1449*	1450	1455*	1458*	1459	1553*	1560	1599*	1636*	1648	1677*	1690	1726*				
RLBAE	003334	1737	2505*	2611*	2728*	2789*	3377*	3384*										
		228#		602*	716	731	782*	1216	1217*	1219	1259*	1279	1377*	1378				
		1516*	1517*	1525*	1525*	1529*	1530	1557*	1574	1598*	1615	1639*	1656	1681*				
RLCS	003324	1704	1705*	2733*	3378*	3385*												
		224#	581	585	594	599*	709	722	770*	780*	786*	893	1128*					
		1130	1134	1250*	1252	1265	1297*	1298	1301	1401*	1404*	1405	1408	1414*				
		1417*	1418	1421	1518*	1591	1594*	1600	1632	1635*	1641	1673	1676*	1683				
		1722	1725*	1730	1947	2377	2385	2393	2404*	2405*	2419*	2434*	2437*	2470*				
		2473*	2502*	2508*	2608*	2613*	2627*	2628*	2629	2636*	2639*	2725*	2730*	3355*				
		3360*	3365*	3391*	3401*	3402												
RLDA	003330	226#	561*	712	724	783*	1172*	1174	1178	1256*	1274	1351*	1352	1480*				
		1482*	1483															

SEP 0135

830#	2009	2264										
196#	586#	1135#	1157#	1179#	1201#	1226#						
186#												
922#	930	935	952#									
217#	2748											
218#	502	547	901	2203	2294	2519	3083	3085				
219#												
140#	2550	2591	2675	2716	2846*							
468#	478#											
188#												
17#	19#	38#	42	44	46	48#	55	81	97	135	456	464
577	592	608	620	1099	3090	3096	3422#					
17#	23#	39#	42	44	46	48#	55	81	97	135	456	464
61	62	63	64	65	66	67	68	69	70	71	72	73
103	104	105	106	107	108	109	110	111	112	113	114	115
467	468	469	470	471	472	473	474	475	476	477	478	479
586	587	588	589	590	591	592	593	594	595	596	597	598
1107	1113	1135	1136	1137	1138	1157	1158	1159	1160	1179	1180	1181
1182	1201	1202	1203	1204	1205	1227	1228	1232	1233	1249	1249	1273
1276	1281	1282	1294	1305	1310	1310	1311	1323	1333	1333	1337	1338
1349	1355	1356	1360	1361	1375	1382	1383	1387	1388	1400	1412	1413
1425	1426	1430	1431	1443	1453	1454	1462	1463	1467	1468	1479	1486
1487	1495	1496	1500	1501	1515	1523	1524	1533	1534	1538	1539	1563
1564	1569	1570	1577	1578	1604	1605	1610	1611	1618	1619	1645	1646
1651	1652	1659	1660	1687	1688	1693	1694	1699	1700	1707	1708	1734
1735	1739	1740	1743	1744	1777	1778	1785	1786	1787	1796	1797	1798
1812	1821	1822	1823	1839	1840	1854	1855	1859	1860	1861	1890	1891
1898	1911	1918	1919	1920	1921	1949	1956	1957	1978	1989	1993	2004
2008	2014	2015	2019	2025	2026	2029	2034	2035	2044	2045	2050	2051
2056	2057	2070	2072	2073	2081	2086	2087	2090	2106	2115	2119	2128
2129	2133	2140	2146	2147	2152	2153	2166	2168	2169	2177	2178	2179
2182	2201	2224	2228	2253	2265	2305	2310	2338	2340	2341	2353	2376
2383	2391	2399	2415	2420	2447	2452	2456	2457	2483	2487	2488	2518
2528	2530	2534	2535	2558	2595	2601	2620	2634	2644	2655	2662	2664
2683	2720	2724	2737	2738	2740	2769	2771	2773	2845	3012	3049	3051
3090	3101	3104	3107	3110	3113	3116	3119	3122	3126	3133	3142	3149
3150	3157	3158	3161	3173	3174	3175	3176	3181	3182	3188	3189	3190
3191	3404	3422	3430									
17#	21#	1978	1993	2008	2106	2119						
17#	22#	66	91	113	143	573						

CVRLBAC -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACV11 306(1063) 08-FEB-82 10:07 PAGE 48-14
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0136

		1823#	1839#	1861#	1899#	1921#	1957#	1989#	2004#	2014#	2019#	2090#	2115#	2128#
		2133#	2182#	2228#	2265#	2353#	2420#	2457#	2488#	2535#	2601#	2664#	2724#	2740#
		2191#												
TAB	025756	2966#	3064#											
TCLK0	036536	2563#	2688#	3334#										
TCLK1	036554	2568#	2693#	3335#										
TCLK2	036567	2573#	2698#	3336#										
TCSR	003344	2578#	2782#	3337#	538*	2370*	2371*	2372	2384*	2392*	2400*	2406	2750	2753*
		2758#	3080											
TCSUO	002776	102	122#											
TCSO	003320	142#	535	2909										
TEMHI	003460	270#	516*	553										
TEMLO	003456			552	2010									
TEMP1	003422			918	920*	921	928*	929						
TEMP2	003412			978*										
TEMP3	003414	252#		960*										
TEMP4	003416	253#		959	969*	971	976*	977*	980					
TEMP5	003420	254#		919*	920	927*	928							
TLMBP	002575	99	118#											
TLMF	003316	141#	533											
TLMIN	002751	100	121#											
TLMOX	023762	2368	2429	2465	2496	2747#								
TLMPID	023766	2547	2672	2749#										
TLM1	021276	2367#												
TLM2	021650	2428#												
TLM3	022054	2464#												
TLM4	022222	2495#												
TLM5	022502	2546#												
TLM6	023364	2671#												
TMPO	003424	256#	807*	810	813*	1820*	1837*	1857*	2038*	2039*	2040*	2041	2066*	2072
		2162*	2168	2327*	2653*	2750*	2953	3116	3133	3157				
TMPI	003426	257#	2067*	2072	2163*	2168	2222*	2296*	2330*	2332*	2337*	2455*	2486*	2533*
		2619*	2643*	2654*	2736*	3141*	3142	3149	3157					
TMPI2	003430	258#	2058*	2058	2071*	2079	2081	2154*	2164	2167*	2175	2177	2223*	2297*
		2329*	2331*	2336*	2563*	2568*	2573*	2688*	2693*	2698*	3138	3140*	3142	3161
TPEAK	036602	2577*	2582*	2702*	2707*	3161	3337#							
TRAP4	010624	569	879#	2280	3395									
TRPFLG	010626	580*	583	800#	1127*	1132	1149*	1154	1171*	1176	1193*	1198	1215*	1224
		1230	2548*	2551	2270*	2276*	2298	2752*	2755					
TSEC	036523	3161												
TSECX	036532	2557												
TURKY	024510	2845												
TSARGC=	000001													
		31												
		315												
TS CODE=	004032	50#												
		101#												
TSERRN=	000264	17#												
		1305#	1555#	1555#	1382#	1412#	1425#	1453#	1462#	1486#	1495#	1523#	1533#	1563#
		1569#	1577#	1604#	1610#	1618#	1645#	1651#	1659#	1687#	1693#	1699#	1707#	1734#
		1739#	1743#	1777#	1785#	1796#	1821#	1854#	1890#	1918#	1949#	1956#	2025#	2034#
		2044#	2050#	2056#	2070#	2086#	2146#	2152#	2166#	2224#	2305#	2310#	2338#	2376#
		2383#	2391#	2399#	2415#	2447#	2452#	2456#	2483#	2487#	2518#	2528#	2530#	2534#
		2558#	2620#	2634#	2644#	2655#	2683#	2737#	2769#	2771#	2773#			
TS EXCP=	000000	61#	62#	63#	65#	98#	102#	105#	111#					
TS FLAG=	000040	1107#	1306#	1333#	1356#	1383#	1413#	1426#	1454#	1463#	1487#	1496#	1524#	1534#

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACV11 306(1063) 08-FEB-82 10:07 H 11
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0137

TSFREE= 040362	1787#	2015#	2029#	2129#	2201#	2253#	2595#	2720#					
TSGMAN= 000000	3422#	3437#											
TSHILI= 000000	17#												
TSLASI= 000001	61#	62#	63#	65#	98#	102#	105#	111#					
TSLOLI= 000000	17#	3422#	3429#										
TSLSYM= 010000	61#	62#	63#	65#	98#	102#	105#	111#					
	17#	66#	61#	112#	143#	573#	588#	604#	616#	628#	1113#	1138#	1160#
	1182#	1204#	1233#	1282#	1311#	1338#	1361#	1388#	1431#	1468#	1501#	1539#	1578#
	1619#	1660#	1708#	1744#	1798#	1823#	1861#	1898#	1921#	1957#	1989#	2004#	2014#
	2090#	2115#	2128#	2182#	2228#	2353#	2420#	2457#	2488#	2535#	2664#	2740#	3191#
TSLTNO= 000043	3422#												
TSNEST= 177777	17#	55#	57#	59#	64#	66#	81#	91#	97#	101#	104#	112#	135#
	143#	456#	460#	464#	573#	577#	588#	592#	604#	608#	616#	620#	628#
	1099#	1113#	1126#	1138#	1148#	1160#	1170#	1182#	1192#	1204#	1212#	1233#	1248#
	1282#	1292#	1294#	1310#	1311#	1321#	1323#	1337#	1338#	1347#	1349#	1360#	1375#
	1371#	1375#	1387#	1388#	1398#	1400#	1430#	1431#	1441#	1443#	1467#	1468#	1477#
	1479#	1500#	1501#	1511#	1515#	1538#	1539#	1552#	1578#	1589#	1619#	1630#	1660#
	1671#	1708#	1718#	1744#	1755#	1798#	1810#	1823#	1835#	1839#	1859#	1861#	1871#
	1898#	1908#	1921#	1936#	1957#	1974#	1978#	1989#	1993#	2004#	2008#	2014#	2019#
	2087#	2090#	2102#	2106#	2115#	2119#	2128#	2133#	2178#	2182#	2196#	2228#	2247#
	2265#	2341#	2353#	2367#	2420#	2428#	2457#	2464#	2488#	2495#	2535#	2546#	2601#
	2662#	2664#	2671#	2724#	2740#	2740#	3096#	3191#					
TSNSO = 000011	55#	57#	59#	64#	66#	81#	91#	97#	101#	104#	112#	135#	143#
	456#	460#	464#	573#	577#	588#	592#	604#	608#	616#	620#	628#	1099#
	1113#	1126#	1138#	1148#	1160#	1170#	1182#	1192#	1204#	1212#	1233#	1248#	1282#
	1292#	1311#	1321#	1310#	1311#	1321#	1323#	1337#	1338#	1347#	1349#	1360#	1375#
	1501#	1511#	1539#	1511#	1515#	1538#	1539#	1552#	1578#	1589#	1619#	1630#	1660#
	1755#	1798#	1810#	1744#	1755#	1798#	1810#	1823#	1835#	1839#	1859#	1861#	1871#
	2090#	2102#	2106#	2115#	2119#	2128#	2133#	2178#	2182#	2196#	2228#	2247#	2265#
	2495#	2546#	2535#	2601#									
TSNS1 = 000003	1294#	1310#	1323#	1337#	1375#	1387#	1400#	1430#	1443#	1467#	1479#		
	1500#	1515#	1538#	1839#	1989#	1993#	2004#	2008#	2014#	2019#	2087#		
	2106#	2115#	2119#	2128#	2178#	2255#	2341#	2601#	2662#	2724#	2738#		
	3429#	3430#											
TSPCNT= 000000	3430#												
TSPTAB= 010065	42#	3437#											
TSPTHV= 000001	17#	3430#	3437										
TSPTNU= 000001	17#												
TSSAVL= 177777	17#												
TSSEGL= 177777	17#	1294#	1306#	1310#	1323#	1333#	1337#	1349#	1356#	1360#	1375#	1383#	1387#
	1400#	1413#	1426#	1430#	1443#	1454#	1463#	1467#	1479#	1487#	1496#	1500#	1515#
	1524#	1534#	1538#	1839#	1859#	2019#	2029#	2087#	2133#	2178#	2265#	2341#	2601#
	2662#	2724#	2738#										
TSSEKO= 010000	1294#	1306#	1310#	1323#	1333#	1337#	1349#	1356#	1360#	1375#	1383#	1387#	1400#
	1413#	1426#	1430#	1443#	1454#	1463#	1467#	1479#	1487#	1496#	1500#	1515#	1524#
	1534#	1538#	1839#	1859#	2019#	2029#	2087#	2133#	2178#	2265#	2341#	2601#	2662#
	2734#	2738#											
TSIZE= 000007	3422#	3437#											
TSSUBN= 000000	17#	1126#	1148#	1170#	1192#	1212#	1248#	1292#	1321#	1347#	1371#	1398#	1441#
	1477#	1511#	1552#	1589#	1630#	1671#	1718#	1755#	1810#	1835#	1871#	1908#	1936#
	1974#	1978#	1993#	2008#	2102#	2106#	2119#	2196#	2247#	2367#	2428#	2464#	2495#
	2546#	2671#											
TSAGL= 177777	17#												
TSTAGN= 010067	17#	55#	81#	97#	135#	456#	464#	577#	588#	608#	620#	1099#	1126#
	1148#	1170#	1192#	1212#	1248#	1292#	1321#	1347#	1356#	1360#	1375#	1383#	1387#
	1552#	1589#	1630#	1671#	1718#	1755#	1810#	1835#	1871#	1908#	1936#	1974#	1978#
	1993#	2008#	2102#	2106#	2119#	2196#	2247#	2367#	2428#	2464#	2495#	2546#	2671#

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 I 11
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0138

T\$TEMP= 000011

3096#	3429#	3430#	61#	62#	63#	65#	66#	9#	98#	100#	102#	103#
56#	58#	60#	61#	62#	63#	65#	66#	9#	98#	100#	102#	103#
105#	106#	107#	108#	109#	110#	111#	112#	114#	116#	117#	118#	119#
616#	617#	618#	619#	620#	621#	622#	623#	624#	625#	626#	627#	628#
1186#	1191#	1204#	1208#	1211#	1225#	1237#	1247#	128#	1286#	1291#	1306#	1310#
1311#	1315#	1327#	1333#	1337#	1338#	1342#	1346#	135#	1360#	1361#	1365#	1370#
1383#	1387#	1385#	1392#	1397#	1413#	1426#	1430#	143#	1435#	1440#	1454#	1463#
1467#	1468#	1472#	1476#	1487#	1496#	1500#	1501#	1505#	1510#	1524#	1534#	1538#
1539#	1543#	1551#	157#	1582#	1583#	1619#	1623#	169#	1660#	1664#	1670#	1708#
1712#	1717#	1744#	174#	1754#	1787#	1798#	182#	189#	1823#	1827#	1834#	1859#
1861#	1865#	1870#	1898#	1903#	1907#	1921#	1925#	1935#	1957#	1961#	1973#	1989#
2004#	2014#	2015#	2029#	203#	203#	2064#	2101#	2115#	2128#	219#	2178#	2182#
2186#	2195#	2201#	2228#	22#	22#	22#	2341#	2353#	2358#	2366#	2420#	2424#
2427#	2457#	2461#	246#	24#	24#	24#	2535#	2539#	2545#	2595#	2662#	2664#
2668#	2670#	2720#	273#	27#	27#	27#	27#	27#	27#	27#	27#	27#

T\$TEST= 000043

17#	1126#	1148#	1170#	11#	1#	1#	1292#	1321#	1347#	1371#	1398#	1441#
1477#	1511#	1552#	1589#	16#	1#	17#	1755#	1810#	1835#	1871#	1908#	1936#
1974#	1978#	1993#	2003#	21#	2#	2119#	2196#	2247#	2367#	2428#	2464#	2495#
2546#	2671#	3090#	3423#	342#	342#	342#	342#	342#	342#	342#	342#	342#

T\$TSM= 177777

17#	465#	467#	469#	471#	47#	47#	513#	522#	573#	586#	587#	588#
603#	604#	612#	616#	6#	6#	6#	1124#	1105#	1106#	1113#	1135#	1136#
1137#	1138#	1157#	1153#	11#	1#	1#	1140#	1181#	1132#	1201#	1202#	1203#
1204#	1226#	1227#	1223#	12#	1#	1#	1149#	1273#	1276#	1281#	1282#	1294#
1305#	1306#	1310#	1311#	13#	1#	1#	1337#	1338#	1349#	1355#	1356#	1360#
1361#	1375#	1382#	1383#	13#	1#	1#	1412#	1413#	1425#	1426#	1430#	1431#
1443#	1453#	1454#	1462#	14#	1#	1#	1479#	1486#	1487#	1495#	1496#	1500#
1501#	1515#	1523#	1524#	15#	1#	1#	1539#	1563#	1564#	1569#	1570#	1577#
1578#	1604#	1605#	1610#	16#	1#	1#	1645#	1646#	1651#	1652#	1659#	1660#
1687#	1688#	1693#	1694#	16#	1#	1#	173#	1734#	1735#	1739#	1740#	1743#
1744#	1777#	1778#	1785#	17#	1#	1#	1747#	1798#	1812#	1821#	1822#	1823#
1839#	1840#	1854#	1855#	18#	1#	1#	1840#	1891#	1898#	1911#	1918#	1919#
1920#	1921#	1949#	1956#	19#	1#	1#	1943#	2004#	2008#	2014#	2015#	2019#
2025#	2026#	2029#	2034#	20#	1#	1#	2050#	2051#	2056#	2057#	2070#	2072#
2073#	2081#	2086#	2087#	20#	1#	1#	2119#	2128#	2129#	2133#	2140#	2146#
2147#	2152#	2153#	2166#	21#	1#	1#	2177#	2178#	2179#	2182#	2201#	2228#
2253#	2265#	2305#	2310#	23#	1#	1#	2341#	2353#	2376#	2383#	2391#	2415#
2420#	2447#	2452#	2456#	24#	1#	1#	2487#	2488#	2518#	2528#	2530#	2535#
2558#	2595#	2601#	2620#	26#	1#	1#	2655#	2662#	2664#	2683#	2720#	2737#
2738#	2740#	2769#	2771#	27#	1#	1#	3012#	3049#	3051#	3101#	3104#	3110#
3113#	3116#	3119#	3122#	31#	1#	1#	3133#	3142#	3149#	3150#	3157#	3173#
3174#	3175#	3176#	3181#	31#	1#	1#	3189#	3190#	3191#	3404#	3404#	3404#

T\$TSTS= 000001

17#	1126#	1148#	1170#	1192#	1212#	1248#	1292#	1321#	1347#	1371#	1398#	1441#
1477#	1511#	1552#	1589#	1630#	1671#	1718#	1755#	1810#	1835#	1871#	1908#	1936#
1974#	2102#	2196#	2247#	2367#	2428#	2464#	2495#	2546#	2671#			

T\$SAU = 010011
T\$SAUT= 010006
T\$SCLE= 010007
T\$SDAT= 010066
T\$SDU = 010010
T\$SHAR= 010000
T\$SHW = 010001
T\$SINI= 010005
T\$MSG = 010063
T\$SPC = 000001
T\$SPRO= 010004
T\$SPTA= 010065

620#	628#
577#	588#
592#	604#
3430#	3436#
608#	616#
55#	66#
81#	91#
464#	573#
3096#	3191#
3429#	3437#
456#	
3429#	3430#

CVF_BAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59 MACY11 30G(1063) 08-FEB-82 10:07 J 11
PAGE 48-17
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0139

TSSRPT= 010012	1099#	1107	1113												
TSSSEG= 010000	1294#	1306	1310#	1323#	1333	1337#	1349#	1356	1360#	1375#	1383	1387#	1400#		
	1413	1426	1430#	1443#	1454	1463	1467#	1479#	1487	1496	1500#	1515#	1524		
	1534	1538#	1839#	1859#	2019#	2029	2087#	2133#	2178#	2265#	2341#	2601#	2662#		
	2724#	2736#													
TSSSOF= 010002	1971#	1989	1993#	2004	2008#	2014	2106#	2115	2119#	2128					
TSSSUB= 010052	135#	143													
TSSSW= 010003	1126#	1138	1148#	1160	1170#	1182	1192#	1204	1212#	1233	1248#	1282	1292#		
TSSTES= 010062	1311	1321#	1338	1347#	1361	1371#	1388	1398#	1431	1441#	1468	1477#	1501		
	1511#	1539	1552#	1578	1599#	1619	1630#	1660	1671#	1708	1718#	1744	1755#		
	1787	1798	1810#	1823	1835#	1861	1871#	1898	1908#	1921	1936#	1957	1974#		
	2015	2090	2102#	2129	2182	2196#	2201	2228	2247#	2253	2353	2357#	2420		
	2428#	2457	2464#	2488	2495#	2535	2546#	2595	2664	2671#	2720	2740			
T1	012114	G													
T10	013332	G													
T11	013436	G													
T12	013664	G													
T13	014050	G													
T14	014216	G													
T15	014410	G													
T16	014612	G													
T17	015036	G													
T18	015262	G													
T19	015532	G													
T2	012200	G													
T20	015746	G													
T21	016160	G													
T22	016246	G													
T23	016414	G													
T24	016550	G													
T25	016634	G													
T26	016762	G													
T26.1	016762														
T26.2	017030														
T26.3	017076														
T27	017652	G													
T27.1	017662														
T27.2	017714														
T28	020306	G													
T29	020502	G													
T3	012264	G													
T30	021276	G													
T31	021650	G													
T32	022054	G													
T33	022222	G													
T34	022502	G													
T35	023364	G													
T4	012350	G													
T5	012434	G													
T6	012556	G													
T7	013020	G													
T8	013140	G													
T9	013244	G													
UAM	000200	G													
UNITST	003322		223#	517*	518*	520	522	536	578	557	1136	1158	1180	1202	1227

MACY11 30G(1063) 08-FEB-82 10:07 K 11 PAGE 48-18
CROSS REFERENCE TABLE -- USER SYMBOLS

K 11

PAGE 48-18

[illegible]

CVRLBAO -- RLV12 DISKLESS MACY11 30G(1063) 08-FEB-82 10:07 L 11
CVRLBA.P11 08-FEB-82 09:59 CROSS REFERENCE TABLE -- MACRO NAMES PAGE 49

SEQ 0141

BCOMPL	466	468	470	472	474	3182									
BEGIN.	211#	1126	1148	1170	1192	1212	1248	1292	1321	1347	1371	1398	1441	1477	1511
	1552	1589	1620	1671	1718	1755	1810	1835	1871	1908	1936	1974	2102	2196	2247
	2367	2428	2464	2495	2546	2671									
BGNAU	620														
BGNAUT	577														
BGNCLN	592														
BGNDU	508														
BGNHRD	55														
BGNHW	81														
BGNINI	464														
BGNMSG	3096														
BGNPRO	456														
BGNPTA	3430														
BGNRPT	1099														
BGNSEG	1294	1323	1349	1375	1400	1443	1479	1515	1839	2019	2133	2265	2601	2724	
BGNSET	3429														
BGNSFT	97														
BGNSUB	1978	1993	2008	2106	2119										
BGNSW	135														
BGNTST	1126	1148	1170	1192	1212	1248	1292	1321	1347	1371	1398	1441	1477	1511	1552
	1589	1630	1671	1718	1755	1810	1835	1871	1908	1936	1974	2102	2196	2247	2367
	2428	2464	2495	2546	2671										
BNCOMP	523														
CKLOOP	1564	1570	1605	1611	1646	1652	1688	1694	1700	1735	1740	1778	1786	1797	1855
	1891	1919	2026	2035	2045	2051	2057	2073	2147	2153	2169	2340			
DESCRI	44														
DEVTYP	46														
DFERR	204#	900	1232	1269	1273	1276	1281	1305	1332	1355	1382	1412	1425	1453	1462
	1486	1495	1523	1533	1563	1569	1577	1604	1610	1618	1645	1651	1659	1687	1693
	1699	1707	1734	1739	1743	1777	1785	1796	1821	1854	1890	1918	1949	1956	2025
	2034	2044	2050	2056	2070	2086	2146	2152	2166	2224	2305	2310	2338	2376	2383
	2391	2399	2415	2447	2452	2456	2483	2487	2518	2528	2530	2534	2558	2620	2634
	2644	2655	2683	2737	2769	2771	2773								
DISPAT	3090														
DISPLA	99														
DOCLN	513	1137	1159	1181	1203	1228	3049	3190							
DODU	587	1136	1158	1180	1202	1227	3189								
DORPT	509														
ENDAU	628														
ENDAUT	588														
ENDCLN	604														
ENDDU	616														
ENDHRD	66														
ENDHW	91														
ENDINI	573														
ENDMSG	3191														
ENDPRO	460														
ENDPTA	3436														
ENDRPT	1113														
ENDSEG	1310	1337	1360	1387	1430	1467	1500	1538	1859	2087	2178	2341	2662	2738	
ENDSET	3437														
ENDSFT	112														
ENDSUB	1989	2004	2014	2115	2128										
ENDSW	143														
ENDTST	1138	1160	1182	1204	1233	1282	1311	1338	1361	1388	1431	1468	1501	1539	1578

	1619	1660	1708	1744	1798	1823	1861	1898	1921	1957	2090	2182	2228	2353	2420
EQUALS	2457	2488	2535	2664	2740										
ERRDF	147														
	900	1232	1269	1273	1276	1281	1305	1332	1355	1382	1412	1425	1453	1462	1486
	1495	1523	1535	1563	1569	1577	1604	1610	1618	1645	1651	1659	1687	1693	1699
	1707	1734	1739	1743	1777	1785	1796	1821	1854	1890	1918	1949	1956	2025	2034
	2044	2050	2056	2070	2086	2146	2152	2166	2224	2305	2310	2338	2376	2383	2391
	2399	2415	2427	2452	2456	2483	2487	2518	2528	2530	2534	2558	2620	2634	2644
ERRSF	2655	2683	2737	2769	2771	2773									
ESCAPE	586	1135	1157	1179	1201	1226									
EXIT	1306	1333	1356	1383	1413	1426	1454	1463	1487	1496	1521	1534			
GETPRI	1107	1787	2015	2029	2129	2201	2253	2495	2720						
GPHARD	3404														
GPRMA	522														
GPRMD	61	62	102												
GPRML	63	65	98	105	111										
HEADER	56	58	60	100	103	106	107	108	109	110					
INLOOP	42														
LASTAD	3181														
MSBYTE	3422														
MSCHEC	42#														
MSCNTO	1107#	1787#	2015#	2029#	2129#	2201#	2253#	2595#	2720#						
	56#	58#	60#	61#	62#	63#	65#	98#	100#	102#	103#	105#	106#	107#	108#
MSCOUN	109#	110#	111#												
	612#	624#	1104#	1105#	1106#	2072#	2081#	2168#	2177#	2845#	3012#	3051#	3101#	3104#	3107#
	3110#	3113#	3116#	3119#	3122#	3126#	3133#	3142#	3149#	3150#	3157#	3158#	3161#	3173#	3174#
MSDATA	3175#	3176#	3188#												
MSDECR	42#	44#	46#												
	66#	91#	112#	143#	460#	573#	588#	604#	616#	628#	1113#	1138#	1160#	1182#	1204#
	1233#	1282#	1310#	1311#	1337#	1338#	1360#	1361#	1387#	1388#	1430#	1431#	1467#	1468#	1500#
	1501#	1538#	1539#	1578#	1619#	1660#	1708#	1744#	1798#	1823#	1859#	1861#	1898#	1921#	1957#
	1989#	2004#	2014#	2087#	2090#	2115#	2128#	2178#	2182#	2228#	2341#	2353#	2420#	2457#	2488#
MSDEFA	2535#	2662#	2664#	2738#	2740#	3191#	3430#								
	56#	58#	60#	61#	62#	63#	65#	98#	100#	102#	103#	105#	106#	107#	108#
MSENDE	109#	110#	111#												
	66#	91#	112#	143#	573#	588#	604#	616#	628#	1113#	1138#	1160#	1182#	1204#	1233#
	1282#	1310#	1311#	1337#	1338#	1360#	1361#	1387#	1388#	1430#	1431#	1467#	1468#	1500#	1501#
	1538#	1539#	1578#	1619#	1660#	1708#	1744#	1798#	1823#	1859#	1861#	1898#	1921#	1957#	1989#
	2004#	2014#	2087#	2090#	2115#	2128#	2178#	2182#	2228#	2341#	2353#	2420#	2457#	2488#	2535#
MSERRI	2662#	2664#	2738#	2740#	3191#										
	586#	900#	1135#	1157#	1179#	1201#	1226#	1232#	1269#	1273#	1276#	1281#	1305#	1332#	1355#
	1382#	1412#	1425#	1462#	1463#	1486#	1495#	1523#	1533#	1563#	1569#	1577#	1604#	1610#	1618#
	1645#	1651#	1659#	1693#	1699#	1707#	1734#	1739#	1743#	1777#	1785#	1796#	1821#	1854#	
	1890#	1918#	1949#	2025#	2034#	2044#	2050#	2056#	2070#	2086#	2146#	2152#	2166#	2224#	
	2305#	2310#	2338#	2363#	2391#	2399#	2415#	2447#	2452#	2456#	2483#	2487#	2518#	2528#	
	2530#	2534#	2558#	2620#	2634#	2644#	2655#	2683#	2737#	2769#	2771#	2773#			
MSESCA	1306#	1333#	1356#	1383#	1413#	1426#	1454#	1463#	1487#	1496#	1524#	1534#			
MSESCS	1306#	1333#	1356#	1383#	1413#	1426#	1454#	1463#	1487#	1496#	1524#	1534#			
MSEXCP	61#	62#	63#	65#	98#	102#	105#	111#							
MSEXIT	1107#	1787#	2015#	2029#	2129#	2201#	2253#	2595#	2720#						
MSEXSE	1107#	1787#	2015#	2029#	2129#	2201#	2253#	2595#	2720#						
MSEX TJ	1107#	1787#	2015#	2029#	2129#	2201#	2253#	2595#	2720#						
MSGEN	42#	44#	46#	55#	66#	81#	91#	97#	112#	135#	143#	456#	464#	573#	577#
	588#	592#	604#	608#	616#	620#	628#	1099#	1113#	1126#	1138#	1148#	1160#	1170#	1182#
	1192#	1204#	1212#	1233#	1248#	1252#	1292#	1310#	1311#	1321#	1337#	1338#	1347#	1360#	1361#
	1371#	1387#	1388#	1398#	1430#	1431#	1441#	1467#	1468#	1477#	1500#	1501#	1511#	1538#	1539#

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 N 11
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0143

	1552#	1578#	1589#	1619#	1630#	1662#	1671#	1708#	1718#	1744#	1755#	1798#	1810#	1823#	1835#
	1855#	1861#	1871#	1898#	1908#	1921#	1936#	1959#	1974#	1978#	1989#	1993#	2004#	2008#	2014#
	2087#	2090#	2102#	2106#	2115#	2119#	2128#	2178#	2182#	2196#	2228#	2247#	2341#	2353#	2367#
	2420#	2428#	2437#	2464#	2488#	2495#	2535#	2546#	2662#	2664#	2671#	2738#	2740#	3090#	3096#
MSGETS	3191#	3422#	3430#	3436#											
	57#	59#	64#	66#	91#	101#	104#	112#	143#	460#	573#	588#	604#	616#	628#
	1113#	1138#	1160#	1182#	1204#	1233#	1282#	1306#	1310#	1311#	1333#	1337#	1338#	1356#	1360#
	1361#	1383#	1387#	1387#	1413#	1424#	1430#	1431#	1454#	1463#	1467#	1463#	1487#	1496#	1500#
	1501#	1534#	1534#	1530#	1539#	1573#	1619#	1660#	1708#	1744#	1798#	1823#	1859#	1861#	1898#
	1921#	1957#	1989#	2004#	2014#	2029#	2087#	2090#	2115#	2128#	2178#	2182#	2228#	2341#	2353#
MSGETT	2420#	2457#	2487#	2535#	2662#	2664#	2738#	2740#	3191#						
	57#	59#	64#	101#	104#	1107#	1306#	1333#	1356#	1383#	1413#	1426#	1454#	1463#	1487#
MSGNGB	1496#	1524#	1534#	1787#	2015#	2029#	2129#	2201#	2253#	2595#	2720#				
	42#	44#	46#	55#	81#	97#	135#	456#	464#	577#	592#	608#	620#	1099#	3090#
MSGNIN	3096#	3422#													
	42#	44#	46#	55#	56#	57#	58#	59#	60#	61#	62#	63#	64#	65#	66#
	81#	97#	98#	99#	100#	101#	102#	103#	104#	105#	106#	107#	108#	109#	110#
	111#	112#	135#	465#	466#	467#	468#	469#	470#	471#	472#	473#	474#	509#	513#
	522#	523#	573#	586#	587#	588#	603#	604#	612#	616#	624#	628#	900#	1104#	1105#
	1106#	1107#	1113#	1135#	1136#	1137#	1138#	1157#	1158#	1159#	1160#	1179#	1190#	1181#	1182#
	1201#	1202#	1203#	1204#	1226#	1227#	1228#	1232#	1233#	1249#	1269#	1273#	1276#	1281#	1282#
	1294#	1305#	1306#	1310#	1311#	1323#	1332#	1333#	1337#	1338#	1349#	1355#	1356#	1360#	1361#
	1375#	1382#	1383#	1387#	1388#	1400#	1412#	1413#	1425#	1426#	1430#	1431#	1443#	1453#	1454#
	1462#	1463#	1467#	1468#	1479#	1486#	1487#	1495#	1496#	1500#	1501#	1515#	1523#	1524#	1533#
	1534#	1538#	1539#	1563#	1564#	1569#	1570#	1577#	1578#	1604#	1605#	1610#	1611#	1618#	1619#
	1645#	1646#	1651#	1652#	1659#	1660#	1687#	1688#	1693#	1694#	1699#	1700#	1707#	1708#	1734#
	1735#	1739#	1740#	1743#	1744#	1777#	1778#	1785#	1786#	1787#	1796#	1797#	1798#	1812#	1821#
	1822#	1823#	1839#	1840#	1854#	1855#	1859#	1860#	1861#	1890#	1891#	1898#	1911#	1918#	1919#
	1920#	1921#	1949#	1956#	1957#	1978#	1989#	1993#	2004#	2008#	2014#	2015#	2019#	2025#	2026#
	2029#	2034#	2035#	2044#	2045#	2050#	2051#	2056#	2057#	2070#	2072#	2073#	2081#	2086#	2087#
	2090#	2106#	2115#	2119#	2128#	2129#	2133#	2140#	2146#	2147#	2152#	2153#	2166#	2168#	2169#
	2177#	2178#	2179#	2182#	2201#	2224#	2228#	2253#	2265#	2305#	2310#	2338#	2340#	2341#	2353#
	2376#	2383#	2391#	2399#	2415#	2420#	2447#	2452#	2456#	2457#	2483#	2487#	2488#	2518#	2528#
	2530#	2534#	2535#	2558#	2595#	2601#	2620#	2634#	2644#	2655#	2662#	2664#	2683#	2720#	2724#
	2737#	2738#	2740#	2769#	2771#	2773#	2845#	3012#	3049#	3051#	3090#	3101#	3104#	3107#	3110#
	3113#	3116#	3119#	3122#	3126#	3133#	3142#	3149#	3150#	3157#	3158#	3161#	3173#	3174#	3175#
MSGNLS	3176#	3181#	3182#	3188#	3189#	3190#	3191#	3404#	3422#	3430#					
MSGNSU	1310#	1357#	1360#	1387#	1430#	1467#	1500#	1538#	1859#	2087#	2178#	2341#	2662#	2738#	
MSGNTA	1978#	1993#	2008#	2106#	2119#										
	66#	91#	112#	143#	573#	588#	604#	616#	628#	1113#	1138#	1160#	1182#	1204#	1233#
	1282#	1311#	1338#	1361#	1388#	1431#	1468#	1501#	1539#	1578#	1619#	1660#	1708#	1744#	1798#
	1813#	1861#	1898#	1921#	1957#	1989#	2004#	2014#	2090#	2115#	2128#	2182#	2228#	2353#	2420#
MSGNTE	2417#	2488#	2535#	2664#	2740#	3191#	3430#	3436#							
	1126#	1148#	1170#	1192#	1212#	1248#	1292#	1321#	1347#	1371#	1398#	1441#	1477#	1511#	1552#
	1519#	1630#	1671#	1718#	1755#	1810#	1835#	1871#	1908#	1936#	1974#	2102#	2196#	2247#	2367#
MSHAPT	2418#	2464#	2495#	2546#	2671#										
MSHINAP	42#														
MSINCR	55#	81#	97#	135#	456#	464#	465#	467#	469#	471#	473#	509#	513#	522#	573#
	577#	587#	587#	588#	592#	603#	604#	608#	612#	618#	620#	624#	628#	900#	1099#
	1104#	1105#	1106#	1113#	1126#	1135#	1136#	1137#	1138#	1148#	1157#	1158#	1159#	1163#	1170#
	1119#	1180#	1181#	1182#	1192#	1201#	1202#	1203#	1204#	1212#	1226#	1227#	1228#	1232#	1233#
	1248#	1249#	1262#	1273#	1276#	1281#	1282#	1292#	1294#	1305#	1306#	1310#	1311#	1321#	1323#
	1312#	1333#	1337#	1338#	1347#	1349#	1355#	1356#	1360#	1361#	1371#	1375#	1382#	1383#	1387#
	1318#	1398#	1400#	1412#	1413#	1425#	1426#	1430#	1431#	1441#	1443#	1453#	1454#	1463#	1467#
	1467#	1468#	1477#	1479#	1486#	1487#	1495#	1496#	1500#	1501#	1511#	1515#	1523#	1524#	1533#

	1534#	1538#	1539#	1552#	1563#	1564#	1569#	1570#	1577#	1578#	1589#	1604#	1605#	1610#	1611#
	1618#	1619#	1619#	1645#	1646#	1651#	1652#	1659#	1660#	1671#	1687#	1688#	1693#	1694#	1699#
	1700#	1707#	1707#	1711#	1734#	1735#	1735#	1740#	1743#	1744#	1758#	1777#	1778#	1785#	1786#
	1787#	1796#	1797#	1797#	1810#	1812#	1821#	1822#	1823#	1835#	1839#	1840#	1854#	1855#	1859#
	1840#	1861#	1871#	1897#	1898#	1898#	1903#	1911#	1918#	1919#	1920#	1921#	1936#	1949#	1956#
	1957#	1974#	1978#	1993#	1993#	2004#	2005#	2014#	2015#	2019#	2023#	2026#	2029#	2034#	2035#
	2044#	2045#	2055#	2051#	2056#	2057#	2070#	2072#	2073#	2081#	2084#	2087#	2090#	2102#	2106#
	2115#	2119#	2128#	2129#	2133#	2140#	2146#	2147#	2152#	2153#	2164#	2166#	2169#	2177#	2178#
	2176#	2182#	2194#	2201#	2224#	2228#	2247#	2253#	2262#	2265#	2271#	2273#	2274#	2341#	2353#
	2347#	2376#	2383#	2391#	2399#	2415#	2420#	2428#	2447#	2452#	2453#	2454#	2454#	2483#	2487#
	2455#	2495#	2518#	2528#	2530#	2534#	2535#	2546#	2558#	2593#	2601#	2603#	2604#	2643#	2655#
	2662#	2664#	2671#	2683#	2720#	2724#	2737#	2738#	2740#	2769#	2777#	2778#	2845#	3012#	3049#
MSLUPO	3051#	3096#	3101#	3104#	3107#	3110#	3113#	3116#	3119#	3122#	3124#	3133#	3142#	3149#	3150#
	3157#	3158#	3161#	3173#	3174#	3175#	3176#	3181#	3188#	3189#	3190#	3191#	3191#	3229#	3430#
	465#	467#	469#	471#	473#	522#	587#	603#	1136#	1158#	1180#	1282#	1227#	1249#	1812#
	1822#	1840#	1860#	1911#	1920#	2140#	2179#	3167#							
MSMCHI	17#														
MSMCHI	17#														
MSPOP	66#	91#	112#	143#	460#	573#	588#	604#	616#	628#	1113#	1138#	1160#	1182#	1204#
	1233#	1282#	1310#	1311#	1337#	1338#	1360#	1361#	1387#	1388#	1436#	1437#	1467#	1468#	1500#
	1501#	1538#	1539#	1578#	1619#	1660#	1708#	1744#	1798#	1823#	1839#	1841#	1898#	1921#	1957#
	1989#	2004#	2014#	2087#	2090#	2115#	2128#	2178#	2182#	2228#	2341#	2353#	2420#	2457#	2488#
MSPRIN	2535#	2662#	2664#	2738#	2740#	3191#									
	612#	624#	1104#	1105#	1106#	2072#	2081#	2168#	2177#	2845#	3012#	3051#	3101#	3104#	3107#
	3110#	3113#	3116#	3119#	3122#	3126#	3133#	3142#	3149#	3150#	3157#	3158#	3161#	3173#	3174#
	3175#	3176#	3188#												
MSPIUSH	55#	81#	97#	135#	456#	464#	577#	592#	608#	620#	1099#	1126#	1148#	1170#	1192#
	1212#	1248#	1292#	1294#	1321#	1323#	1347#	1349#	1371#	1375#	1398#	1400#	1441#	1443#	1477#
	1479#	1511#	1515#	1552#	1589#	1630#	1671#	1718#	1755#	1810#	1835#	1839#	1871#	1908#	1936#
	1974#	1978#	1993#	2008#	2019#	2102#	2106#	2119#	2133#	2196#	2247#	2265#	2367#	2428#	2464#
	2495#	2546#	2601#	2671#	2724#	3096#									
MSPUT	612#	624#	1104#	1105#	1106#	2072#	2081#	2168#	2177#	2845#	3012#	3051#	3101#	3104#	3107#
	3110#	3113#	3116#	3119#	3122#	3126#	3133#	3142#	3149#	3150#	3157#	3158#	3161#	3173#	3174#
	3175#	3176#	3188#												
MSPUT1	612#	624#	1104#	1105#	1106#	2072#	2081#	2168#	2177#	2845#	3012#	3051#	3101#	3104#	3107#
	3110#	3113#	3116#	3119#	3122#	3126#	3133#	3142#	3149#	3150#	3157#	3158#	3161#	3173#	3174#
	3175#	3176#	3188#												
MSRADI	56#	58#	60#	61#	62#	63#	65#	98#	100#	102#	103#	105#	106#	107#	108#
	109#	110#	111#												
MSRNRO	522#	3404#													
MSSETS	55#	81#	97#	135#	456#	464#	577#	592#	608#	620#	1099#	1126#	1148#	1170#	1192#
	1212#	1248#	1292#	1294#	1321#	1323#	1347#	1349#	1371#	1375#	1398#	1400#	1441#	1443#	1477#
	1479#	1511#	1515#	1552#	1589#	1630#	1671#	1718#	1755#	1810#	1835#	1839#	1871#	1908#	1936#
	1974#	1978#	1993#	2008#	2019#	2102#	2106#	2119#	2133#	2196#	2247#	2265#	2367#	2428#	2464#
	2495#	2546#	2601#	2671#	2724#	3096#									
MSVC	465#	467#	469#	471#	473#	509#	513#	522#	573#	586#	587#	589#	603#	604#	612#
	616#	624#	628#	900#	1104#	1105#	1106#	1107#	1113#	1135#	1136#	1137#	1138#	1157#	1158#
	1159#	1160#	1179#	1180#	1181#	1182#	1201#	1202#	1203#	1204#	1226#	1227#	1228#	1232#	1233#
	1249#	1269#	1273#	1276#	1281#	1282#	1294#	1305#	1306#	1310#	1311#	1323#	1332#	1333#	1337#
	1338#	1349#	1355#	1356#	1360#	1361#	1375#	1382#	1383#	1387#	1388#	1400#	1412#	1413#	1425#
	1426#	1430#	1431#	1443#	1453#	1454#	1462#	1463#	1467#	1468#	1479#	1486#	1487#	1495#	1496#
	1500#	1501#	1515#	1523#	1524#	1533#	1534#	1538#	1539#	1563#	1564#	1569#	1570#	1577#	1578#
	1604#	1605#	1610#	1611#	1618#	1619#	1645#	1646#	1651#	1652#	1659#	1660#	1687#	1688#	1693#
	1694#	1699#	1700#	1707#	1708#	1734#	1735#	1735#	1740#	1743#	1744#	1758#	1777#	1778#	1785#
	1787#	1796#	1797#	1798#	1812#	1821#	1822#	1823#	1835#	1839#	1840#	1854#	1855#	1859#	1861#
	1890#	1891#	1898#	1911#	1918#	1919#	1920#	1921#	1949#	1956#	1957#	1978#	1989#	1993#	2004#

CVRLBAO -- RLV12 DISKLESS
CVRLBA.P11 08-FEB-82 09:59

MACY11 30G(1063) 08-FEB-82 10:07 C 12
PAGE 49-4
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0145

MSLAB

MSSTL

MSWORD

MSXFER

2008#	2014#	2015#	2019#	2025	2026#	2029#	2034	2035#	2044	2045#	2050	2051#	2056	2057#
2070#	2072#	2073#	2081#	2086	2087#	2090#	2106#	2115#	2119#	2128#	2139#	2140#	2140#	2146#
2147#	2152#	2153#	2166#	2168#	2169#	2177#	2178#	2179#	2182#	2183#	2184#	2185#	2186#	2187#
2305#	2310#	2338#	2340#	2341#	2342#	2343#	2344#	2345#	2346#	2347#	2348#	2349#	2350#	2351#
2457#	2483#	2487#	2488#	2489#	2490#	2491#	2492#	2493#	2494#	2495#	2496#	2497#	2498#	2499#
2655#	2662#	2664#	2683#	2720#	2721#	2722#	2723#	2724#	2725#	2726#	2727#	2728#	2729#	2730#
3051#	3101#	3104#	3107#	3110#	3111#	3112#	3113#	3114#	3115#	3116#	3117#	3118#	3119#	3120#
3158#	3161#	3173#	3174#	3175#	3176#	3177#	3178#	3179#	3180#	3181#	3182#	3183#	3184#	3185#
465#	467#	469#	471#	473#	509#	513#	522#	523#	586#	587#	588#	603#	604#	612#
616#	624#	628#	900#	1104#	1105#	1106#	1113#	1133#	1136#	1137#	1138#	1157#	1158#	1159#
1160#	1179#	1180#	1181#	1182#	1201#	1202#	1203#	1204#	1226#	1227#	1228#	1232#	1233#	1249#
1269#	1273#	1276#	1281#	1282#	1294#	1305#	1306#	1310#	1311#	1323#	1332#	1333#	1337#	1338#
1349#	1355#	1356#	1360#	1361#	1375#	1382#	1383#	1387#	1388#	1400#	1412#	1413#	1425#	1426#
1430#	1431#	1443#	1453#	1454#	1462#	1463#	1467#	1468#	1479#	1486#	1487#	1495#	1496#	1500#
1501#	1515#	1523#	1524#	1533#	1534#	1538#	1539#	1563#	1564#	1569#	1570#	1577#	1578#	1604#
1605#	1610#	1611#	1618#	1619#	1645#	1646#	1651#	1652#	1659#	1660#	1687#	1688#	1693#	1694#
1699#	1700#	1707#	1708#	1734#	1735#	1739#	1740#	1743#	1744#	1777#	1778#	1785#	1786#	1787#
1796#	1797#	1798#	1812#	1821#	1822#	1823#	1839#	1840#	1854#	1855#	1859#	1860#	1861#	1890#
1891#	1898#	1911#	1918#	1919#	1920#	1921#	1949#	1956#	1957#	1978#	1989#	1993#	2004#	2008#
2014#	2015#	2019#	2025#	2026#	2029#	2034#	2035#	2044#	2045#	2050#	2051#	2056#	2057#	2070#
2072#	2073#	2081#	2086#	2087#	2090#	2106#	2115#	2119#	2128#	2129#	2133#	2140#	2146#	2147#
2152#	2153#	2166#	2168#	2169#	2177#	2178#	2179#	2182#	2201#	2224#	2228#	2253#	2265#	2305#
2310#	2338#	2340#	2341#	2353#	2376#	2383#	2391#	2399#	2415#	2420#	2447#	2452#	2456#	2457#
2483#	2487#	2488#	2518#	2528#	2530#	2534#	2535#	2558#	2595#	2601#	2620#	2634#	2644#	2655#
2662#	2664#	2683#	2720#	2721#	2722#	2723#	2740#	2769#	2771#	2773#	2845#	3012#	3049#	3051#
3101#	3104#	3107#	3110#	3113#	3116#	3119#	3122#	3126#	3133#	3142#	3149#	3150#	3157#	3158#
3161#	3173#	3174#	3175#	3176#	3181#	3188#	3189#	3190#	3191#	3404#				
465#	467#	469#	471#	473#	509#	513#	522#	523#	586#	587#	588#	603#	604#	612#
616#	624#	628#	900#	1104#	1105#	1106#	1113#	1133#	1136#	1137#	1138#	1157#	1158#	1159#
1160#	1179#	1180#	1181#	1182#	1201#	1202#	1203#	1204#	1226#	1227#	1228#	1232#	1233#	1249#
1269#	1273#	1276#	1281#	1282#	1294#	1305#	1306#	1310#	1311#	1323#	1332#	1333#	1337#	1338#
1349#	1355#	1356#	1360#	1361#	1375#	1382#	1383#	1387#	1388#	1400#	1412#	1413#	1425#	1426#
1430#	1431#	1443#	1453#	1454#	1462#	1463#	1467#	1468#	1479#	1486#	1487#	1495#	1496#	1500#
1501#	1515#	1523#	1524#	1533#	1534#	1538#	1539#	1563#	1564#	1569#	1570#	1577#	1578#	1604#
1605#	1610#	1611#	1618#	1619#	1645#	1646#	1651#	1652#	1659#	1660#	1687#	1688#	1693#	1694#
1699#	1700#	1707#	1708#	1734#	1735#	1739#	1740#	1743#	1744#	1777#	1778#	1785#	1786#	1787#
1796#	1797#	1798#	1812#	1821#	1822#	1823#	1839#	1840#	1854#	1855#	1859#	1860#	1861#	1890#
1891#	1898#	1911#	1918#	1919#	1920#	1921#	1949#	1956#	1957#	1978#	1989#	1993#	2004#	2008#
2014#	2015#	2019#	2025#	2026#	2029#	2034#	2035#	2044#	2045#	2050#	2051#	2056#	2057#	2070#
2072#	2073#	2081#	2086#	2087#	2090#	2106#	2115#	2119#	2128#	2129#	2133#	2140#	2146#	2147#
2152#	2153#	2166#	2168#	2169#	2177#	2178#	2179#	2182#	2201#	2224#	2228#	2253#	2265#	2305#
2310#	2338#	2340#	2341#	2353#	2376#	2383#	2391#	2399#	2415#	2420#	2447#	2452#	2456#	2457#
2483#	2487#	2488#	2518#	2528#	2530#	2534#	2535#	2558#	2595#	2601#	2620#	2634#	2644#	2655#
2662#	2664#	2683#	2720#	2721#	2722#	2723#	2740#	2769#	2771#	2773#	2845#	3012#	3049#	3051#
3101#	3104#	3107#	3110#	3113#	3116#	3119#	3122#	3126#	3133#	3142#	3149#	3150#	3157#	3158#
3161#	3173#	3174#	3175#	3176#	3181#	3188#	3189#	3190#	3191#	3404#				
42#	56#	77#	58#	59#	60#	61#	62#	63#	64#	65#	98#	99#	100#	101#
102#	103#	104#	105#	106#	107#	108#	109#	110#	111#	112#	113#	114#	115#	116#
117#	118#	119#	120#	121#	122#	123#	124#	125#	126#	127#	128#	129#	130#	131#
132#	133#	134#	135#	136#	137#	138#	139#	140#	141#	142#	143#	144#	145#	146#
147#	148#	149#	150#	151#	152#	153#	154#	155#	156#	157#	158#	159#	160#	161#
162#	163#	164#	165#	166#	167#	168#	169#	170#	171#	172#	173#	174#	175#	176#
177#	178#	179#	180#	181#	182#	183#	184#	185#	186#	187#	188#	189#	190#	191#
192#	193#	194#	195#	196#	197#	198#	199#	200#	201#	202#	203#	204#	205#	206#
207#	208#	209#	210#	211#	212#	213#	214#	215#	216#	217#	218#	219#	220#	221#
222#	223#	224#	225#	226#	227#	228#	229#	230#	231#	232#	233#	234#	235#	236#
237#	238#	239#	240#	241#	242#	243#	244#	245#	246#	247#	248#	249#	250#	251#
252#	253#	254#	255#	256#	257#	258#	259#	260#	261#	262#	263#	264#	265#	266#
267#	268#	269#	270#	271#	272#	273#	274#	275#	276#	277#	278#	279#	280#	281#
282#	283#	284#	285#	286#	287#	288#	289#	290#	291#	292#	293#	294#	295#	296#
297#	298#	299#	300#	301#	302#	303#	304#	305#	306#	307#	308#	309#	310#	311#
312#	313#	314#	315#	316#	317#	318#	319#	320#	321#	322#	323#	324#	325#	326#
327#	328#	329#	330#	331#	332#	333#	334#	335#	336#	337#	338#	339#	340#	341#
342#	343#	344#	345#	346#	347#	348#	349#	350#	351#	352#	353#	354#	355#	356#
357#	358#	359#	360#	361#	362#	363#	364#	365#	366#	367#	368#	369#	370#	371#
372#	373#	374#	375#	376#	377#	378#	379#	380#	381#	382#	383#	384#	385#	386#
387#	388#	389#	390#	391#	392#	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#	440#	441#	442#	443#	444#	445#	446#
447#	448#	449#	450#	451#	452#	453#	454#	455#	456#	457#	458#	459#	460#	461#
462#	463#	464#	465#	466#	467#	468#	469#	470#	471#	472#	473#	474#	475#	476#
477#	478#	479#	480#	481#	482#	483#	484#	485#	486#	487#	488#	489#	490#	491#
492#	493#	494#	495#	496#	497#	498#	499#	500#	501#	502#	503#	504#	505#	506#
507#	508#	509#	510#	511#	512#	513#	514#	515#	516#	517#	518#	519#	520#	521#
522#	523#	524#	525#	526#	527#	528#	529#	530#	531#	532#	533#	534#	535#	536#
537#	538#	539#	540#	541#	542#	543#	544#	545#	546#	547#	548#	549#	550#	551#
552#	553#	554#	555#	556#	557#	558#	559#	560#	561#	562#	563#	564#	565#	566#
567#	568#	569#	570#	571#	572#	573#	574#	575#	576#	577#	578#	579#	580#	581#
582#	583#	584#	585#	586#	587#	588#	589#	590#	591#	592#	593#	594#	595#	596#
597#	598#	599#	600#	601#	602#	603#	604#	605#	606#	607#	608#	609#	610#	611#
612#	613#	614#	615#	616#	617#	618#	619#	620#	621#	622#	623#	624#	625#	626#
627#	628#	629#	630#	631#	63									

CVRLBAO -- RLV12 DISKLESS MACV11 306(1063) 08-FEB-82 10:07 D 12
 CVRLBA.P11 08-FEB-82 09:59 CROSS REFERENCE TABLE -- MACRO NAMES PAGE 49-5

SEQ 0146

POINTE	41														
PRINTB	2081	2177	3101	3104	3107	3110	3113	3116	3119	3122	3126	3133	3142	3173	3174
PRINTF	3175	3176													
PRIN'Y	612	624	1104	1105	1106	2845	3012	3051	3188						
READER	2072	2168	3149	3150	3157	3158	3161								
SETPRI	465	467	469	471	473										
SFERR	603	1249	1812	1822	1840	1860	1911	1920	2140	2179					
STARS	199#	586	1135	1157	1179	1201	1226								
	1120	1125	1142	1147	1164	1169	1186	1191	1208	1211	1237	1247	1286	1291	1315
	1320	1342	1346	1365	1370	1392	1397	1435	1440	1472	1476	1505	1510	1543	1551
	1582	1588	1623	1629	1664	1670	1712	1717	1748	1754	1802	1809	1827	1834	1865
	1870	1903	1907	1925	1935	1961	1973	2094	2101	2186	2195	2232	2245	2358	2366
	2424	2427	2461	2463	2492	2494	2539	2545	2668	2670					
SVC	16#	17													
XFER	59	64	1107#	1787#	2015#	2029#	2129#	2201#	2253#	2595#	2720#				
XFERF	57	101													
XFERT	104														

. ABS. 040362 000 CON RO REL GBL D

ERRORS DETECTED: 0

CVRLBA.BIC CVRLBA.LST/LI:TOC/CRF=SVC33/ML,CVRLBA.P11

RUN-TIME: 36 37 4 SECONDS

RUN-TIME RATIO: 327/78=4.1

CORE USED: 28K (55 PAGES)