

RLV11, RL01,
RL02

RLV11, RL01 DISKLESS
CVRLACO

AH-B108C MC
FICHE 1 OF 1

MAY 1983
COPYRIGHT © 78-83
MADE IN USA

00000000

.MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 1

CVRLAC.P11 31-AUG-82 11:25

SEQ 0001

.REM 8

IDENTIFICATION

PRODUCT CODE: AC-B107C-MC

PRODUCT NAME: CVRLAC0 RLV11 RL01 DSKLS

PRODUCT DATE: SEPTEMBER 1982

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: W. HEAVEY

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

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

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

COPYRIGHT (C) 1978,1983 DIGITAL EQUIPMENT CORPORATION

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT FOR THE RLV11 RL01 DISKLESS TEST

1.1.1 STRUCTURE OF PROGRAM

THE DIAGNOSTIC SUPERVISOR > REV C HAS BEEN RELEASED SEPARATED FROM THIS PROGRAM. ONE SHOULD REFER TO THE XXDP+/SUPP USERS MANUAL (AC-F348_-MC) FOR SUPERVISOR LOADING AND OPERATING INSTRUCTIONS.

THIS DIAGNOSTIC WITH THE SUPERVISOR OCCUPIES UP TO 14.5K WORDS OF MEMORY AND IS COMPATIBLE WITH BOTH XXDP+ AND ACT. IT CAN BE RUN IN STANDALONE UNDER XXDP+ AND CAN BE CHAINED UNDER XXDP+,ACT AND RUN ON APT AND ACT ON APT.

WHEN THIS DIAGNOSTIC IS STARTED AT ADDRESS 200, CONTROL GOES FIRST TO THE SUPERVISOR PORTION, WHICH WILL ASK CERTAIN 'HARD CORE' QUESTIONS ABOUT THE ENVIRONMENT. THEN IT WILL ENTER COMMAND MODE, INDICATED BY A PROMPT CHARACTER. AT COMMAND MODE THE OPERATOR MAY ENTER ANY OF SEVERAL COMMANDS AS DESCRIBED IN THE XXDP+/SUPR USERS MANUAL (AC-F348_-MC).

1.1.2 DIAGNOSTIC INFORMATION

THE RLV11 RL01 DISKLESS TEST IS A LSI-11(PDP-11) BASED PROGRAM THAT WILL TEST THE CONTROLLER.

RLV11 CONTROLLER

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

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

PDP-11/LSI-11 PROCESSOR WITH 16K OR MORE OF CORE
CONSOLE DEVICE (LA30,LA36,VT50,ETC.)
RLV11 CONTROLLER(S) (1-8)
KW11P, KW11L (OPTIONAL)
LINEPRINTER(OPTIONAL)

1.2.2 SOFTWARE REQUIREMENTS

1.3 RELATED DOCUMENTS AND STANDARDS

RL01 USERS MANUAL (EK-RL01-UG-PRE)
CHQUSB XXDP+/SUPR USER MAN (AC-F348 -MC)
FOR DIAGNOSTIC SUPERVISOR COMMAND INSTRUCTIONS.

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

1.5 ASSUMPTIONS

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

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURES

2.1.1 LOADING PROCEDURES

FOR LOADING AND OPERATING INSTRUCTIONS, PLEASE REFER TO
CHQUSB XXDP+/SUPR USER MAN (AC-F348 -MC).
ISSUE THE COMMAND 'R CVRLAC'. THE XXDP+ MONITOR WILL LOAD
THE DIAGNOSTIC AND THE SUPERVISOR FILE HSAA??.SYS AND THEN
GIVE CONTROL TO THE SUPERVISOR.

2.1.2 STARTING PROCEDURES

THE PROGRAM STARTS AT LOCATION 200. USE STANDARD DEC PROCEDURES
TO START THE PROGRAM.
PLEASE REFER TO CHQUSB XXDP+/SUPR MAN (AC-F348 -MC) FOR
SUPERVISOR STARTUP COMMANDS.

2.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION

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

- A) LOAD THE DIAGNOSTIC
- B) START AT ADDRESS 200
- C) ANSWER THE HARDCORE QUESTIONS
- D) RECEIVE PROMPT

E) ENTER STA<CR>
F) ANSWER HARDWARE AND SOFTWARE QUESTIONS
G) GET END OF PASS MESSAGES OR ERROR MESSAGES
H) TO END EXECUTION, ENTER CONTROL/C

2.2 SPECIAL ENVIRONMENTS

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

2.3 PROGRAM OPTIONS

2.3.1 START COMMAND

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

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

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

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

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

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

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

HOE	HALT ON ERROR, CAUSING COMMAND MODE TO BE ENTERED WHEN AN ERROR IS ENCOUNTERED
LOE	LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAINING THE ERROR
IER	INHIBIT ERROR REPORTING

216 IBE INHIBIT BASIC ERROR REPORTS
217 IXE INHIBIT EXTENDED ERROR REPORTS
218 PRI DIRECT ALL MESSAGES TO A LINE PRINTER
219 PNT PRINT NUMBER OF TEST BEING EXECUTED
220 BOE BELL ON ERROR
221 UAM RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS
222 ISR INHIBIT STATISTICAL REPORTS
223 IDR INHIBIT DROPPING OF UNITS BY DIAGNOSTIC

224
225 THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0
226 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH
227 IS NOT GIVEN ALL FLAGS ARE CLEARED.
228

229
230 2.3.1.4 END OF PASS SWITCH (/EOP:<INCR>)

231 <INCR> IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF PASSES)
232 IT IS DESIRED THAT THE END OF PASS MESSAGE BE PRINTED. THE DEFAULT IS
233 AT THE END OF EVERY PASS.
234

235
236 2.3.2 CONTROL CHARACTERS
237 -----
238

239
240 A CONTROL C (^C) ENTERED VIA THE CONSOLE DEVICE DURING THE EXECUTION
241 OF A DIAGNOSTIC CAUSES A RETURN TO THE DIAGNOSTIC SUPERVISOR
242 COMMAND MODE.
243

244 A CONTROL Z (^Z) ENTERED WITHIN ONE OF THE THREE OPERATOR DIALOGS
245 (HARDCORE, HARDWARE, OR SOFTWARE QUESTIONS) CAUSES TO DEFAULT
246 VALUES TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.
247

248 A CONTROL O (^O) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC
249 CAUSES ALL CONSOLE OUTPUT TO BE SUPPRESSED FOR THE REMAINDER
250 OF THE DIAGNOSTIC OR UNTIL ANOTHER CONTROL O IS TYPED.
251

252
253 2.3.4 HARDWARE PARAMETERS
254 -----
255

256 THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND. THE
257 VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT
258 VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONSE.
259

260 11/23 PROCESSOR (L) Y?

261
262 ANSWER YES(Y) IF YOU HAVE AN 11/23 LSI-11 BUS PROCESSOR,
263 ANSWER NO (N) IF YOU HAVE AN LSI-11(11/03L ETC.) PROCESSOR.
264

265 BUS ADDRESS (O) 174400?

266
267 ANSWER WITH THE BUS ADDRESS OF THE CONTROLLER.
268

269 VECTOR (O) 330?

270
271 ANSWER WITH THE INTERRUPT VECTOR OF THE CONTROLLER.

DRIVE (0) 0?

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

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

BR LEVEL(0) 5?

ANSWER WITH THE INTERRUPT PRIORITY OF THE CONTROLLER.

2.3.5 SOFTWARE PARAMETERS

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

CHANGE SW?

IS ANSWERED YES(Y). THE QUESTIONS ARE:

DROP ON ERROR LIMIT (L) Y?

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

ANSWER Y OR N

ERROR LIMIT (D) 10?

NUMBER OF ERRORS ALLOWED BEFORE DROPPING UNIT.

ANSWER 1 TO 65K

AUTOSIZE (L) N?

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

ANSWER Y OR N

2.3.6 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

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

AS SOON AS THE QUESTION '# UNITS?' IS ANSWERED (WITH THE NUMBER N, SAY) SPACE IN CORE IS ALLOCATED FOR N P-TABLES. ALL OF THE P-TABLES

ARE OF THE SAME FORMAT, AND THERE IS A ONE-TO ONE CORRESPONDENCE BETWEEN THE HARDWARE PARAMETER QUESTIONS AND THE SLOTS IN THE P-TABLE FORMAT.

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

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

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

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

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

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

THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:

UNITS (D) ? 64

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

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

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

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

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

2.4 EXECUTION TIMES

ONE PASS OF THE PROGRAM TAKES APPROXIMATELY < 30 SECONDS.

3.0 ERROR INFORMATION

3.1 ERROR REPORTING

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

CVRL? XXX ERR YYYYY TST ZZZ SUB PPP PC: RRRRRR

WHERE:

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

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

EXAMPLE:

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

REGISTER DESCRIPTIONS CAN BE FOUND IN SECTION 5.0.
CS:CONTROL AND STATUS REGISTER

BA:BUS ADDRESS REGISTER
DA:DISK ADDRESS REGISTER
MP:MULTIPURPOSE REGISTER

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

EXAMPLE: DR>STA/FLAG:IXE OR DR>RES/FLAG:IXE

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

3.2 ERROR HALTS

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

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS

THIS PROGRAM WILL NOT GIVE ANY PERFORMANCE REPORTS.

4.2 PROGRESS REPORTS

THIS PROGRAM WILL NOT GIVE ANY PROGRESS REPORTS.

5.0 DEVICE INFORMATION TABLES

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

RLCS - CONTROL AND STATUS REGISTER (XXXXX0)

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

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

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

BIT 0 - DRIVE READY

RLBA - BUS ADDRESS REGISTER (XXXXX2)

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

RLDA - DISK ADDRESS REGISTER (XXXXX4)

FOR READ/WRITE FUNCTIONS

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

FOR SEEK FUNCTION

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

FOR GET STATUS FUNCTION

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

RLMP - MULTIPURPOSE REGISTER

FOR READ/WRITE FUNCTION

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

FOR READ AFTER MAINTENANCE FUNCTION

BIT 15-0

FIRST RLMP: CRC OF STARTING DISK ADDRESS VALUE+3

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

FOR GET STATUS FUNCTION

HAS DRIVE STATUS

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

6.0 TEST SUMMARIES

TEST 1 - RLCS WRITE ADDRESSABILITY

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

TEST 2 - RLBA WRITE ADDRESSABILITY

608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663

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

TEST 3 - RLDA WRITE ADDRESSABILITY

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

TEST 4 - RLMP WRITE ADDRESSABILITY

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

TEST 5 - RLCS READ ADDRESSABILITY

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

TEST 6 - RLBA READ ADDRESSABILITY

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

TEST 7 - RLDA READ ADDRESSABILITY

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

TEST 8 - RLMP READ ADDRESSABILITY

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

TEST 9 - BUS RESET OF RLCS

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

TEST 10 - BUS RESET OF RLBA

664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719

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

TEST 11 - BUS RESET OF RLDA

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

TEST 12 - READ WRITE OF RLCS

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

TEST 13 - READ WRITE OF RLBA

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

TEST 14 - READ WRITE OF RLDA

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

TEST 15 - BIS OF RLCS

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

TEST 16 - BIC OF RLCS

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

TEST 17 - BIS OF RLBA

THIS TEST WILL USE THE 11 INSTRUCTION 'BIS' TO SHOW THAT A
READ-MODIFY-WRITE SEQUENCE OF THE RLBA WORKS. BITS 15-0 ARE
BIT SET USING GROWING AND WALKING 0'S AND 1'S. BIT 0 CAN SET
ON A RLV11.

720
721
722 TEST 18 - BIC OF RLBA
723

724 THIS TEST WILL USE THE 11 INSTRUCTION 'BIC' TO SHOW THAT A
725 READ-MODIFY-WRITE SEQUENCE OF THE RLBA WORKS. BITS 15-0 ARE
726 BIT CLEARED USING GROWING AND WALKING 0'S AND 1'S.
727

728
729 TEST 19 - BIS OF RLDA
730

731 THIS TEST WILL USE THE 11 INSTRUCTION 'BIS' TO SHOW THAT A
732 READ-MODIFY-WRITE SEQUENCE OF THE RLDA WORKS. BITS 15-0 ARE
733 BIT SET USING GROWING AND WALKING 0'S AND 1'S.
734

735
736 TEST 20 - BIC OF RLDA
737

738 THIS TEST WILL USE THE 11 INSTRUCTION 'BIC' TO SHOW THAT A
739 READ-MODIFY-WRITE SEQUENCE OF THE RLDA WORKS. BITS 15-0 ARE
740 BIT CLEARED USING GROWING AND WALKING 0'S AND 1'S.
741

742
743 TEST 21 - BUS RESET OF RLCS
744

745 THIS TEST WILL VERIFY THAT THE BUS RESET OF THE PROCESSOR WILL
746 CLEAR ALL BITS OF THE RLCS WITH THE EXCEPTION OF BIT 7 (CON-
747 TROLLER READY), BIT 0 (DRIVE READY) AND BIT 15 (COMPOSITE ERROR)
748 WHICH WILL BE SET IF BIT 14 (DRIVE ERROR) IS SET.
749

750
751 TEST 22 - BUS RESET OF RLBA
752

753 THIS TEST WILL VERIFY THAT THE BUS RESET OF THE PROCESSOR WILL
754 CLEAR ALL BITS OF THE RLBA.
755

756
757 TEST 23 - BUS RESET OF RLDA
758

759 THIS TEST WILL VERIFY THAT THE BUS RESET OF THE PROCESSOR WILL
760 CLEAR ALL BITS OF THE RLDA.
761

762
763 TEST 24 - UNIQUENESS OF RLCS
764

765 THIS TEST WILL VERIFY THAT WHEN THE RLCS (XXXXX0) IS ADDRESSED
766 ONLY THAT REGISTER IS EFFECTED. BOTH THE RLBA AND THE RLDA
767 ARE SET UP WITH KNOWN DATA, THE RLDA IS WRITTEN, THEN THE RLBA
768 AND RLDA ARE VERIFIED THAT THEY DID NOT CHANGE.
769

770
771 TEST 25 - UNIQUENESS OF RLBA
772

773 THIS TEST WILL VERIFY THAT WHEN THE RLBA (XXXXX2) IS ADDRESSED
774 ONLY THAT REGISTER IS EFFECTED. BOTH THE RLCS AND RLDA ARE
775 WRITTEN WITH KNOWN DATA, THE RLBA IS WRITTEN, THEN THE RLCS

776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831

AND RLDA ARE VERIFIED THAT THEY DID NOT CHANGE.

TEST 26 - UNIQUENESS OF RLDA

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

TEST 27 - UNIQUENESS OF RLMP

THIS TEST WILL VERIFY THAT WHEN THE RLMP (XXXXX6) IS ADDRESSED ONLY THAT REGISTER IS EFFECTED. THE RLCS, RLBA AND RLDA ARE WRITTEN WITH KNOWN DATA, THE RLMP IS WRITTEN, THEN THE RLCS, RLBA AND RLDA ARE VERIFIED THAT THEY DID NOT CHANGE.

TEST 28 - RLV11 MAINT. FORCED OPI TEST, LESS THAN 510 WORDS

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

TEST 29 - RLV11 MAINT. FORCED OPI TEST, MORE THAN 511 WORDS

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

TEST 30 - RLV11 MAINT. FORCED OPI TEST - INTERRUPT MODE

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

TEST 31 - RLV11 OPI TIMEOUT TEST

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

TEST 32 - TEST RLV11 MAINT. FUNCTION -FLAG MODE

PERFORM RLV11 MAINTENANCE FUNCTION 0 (FLAG MODE) AND CHECK DA AND BA REGISTERS FOR PROPER INCREMENT. CHECK THE SERIAL WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER THE CRC OF THE DA+3 AND THE CRC OF THE CRC OF THE DA+4 AND COMPARING TO EXPECTED RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE FIFO INTO BUF2 MEMORY FOR PROPER DATA. CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR A VALUE=123456 TO INSURE THAT THE DATA TRANSFER IN MAINTENANCE MODE WAS

832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887

NOT MORE THAN 255 WORDS.

TEST 33 - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

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

TEST 34 - RLV11 FIFO ADDRESS TEST

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

TEST 35 - RLV11 FIFO ADDRESS COMPLEMENT TEST

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

TEST 36 - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INTERRUPT MODE

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

TEST 37 - TEST RLV11 MAINT. WITH RANDOM DATA -INTERRUPT MODE

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

888
889
890
891
892
893
894
895
896
897 000000
898 000000
899 000000
900 002000
901
902
903 002000
904
905
906 002000
907
908 002000
909 002000 103
910 002001 126
911 002002 122
912 002003 114
913 002004 101
914 002005 000
915 002006 000
916 002007 000
917 002010 103
918 002011 060
919 002012 000000
920 002014 000036
921 002016 027436
922 002020 027574
923 002022 013574
924 002024 013610
925 002026 030520
926 002030 000000
927 002032 000000
928 002034 000000
929 002036 000000
930 002040 013620
931 002042 000340
932 002044 000000
933 002046 000000
934 002050 003
935 002051 003
936 002052 000000
937 002054 000000
938 002056 000000
939 002060 002122
940 002062 000000
941 002064 000000
942 002066 000000
943 002070 014510

7.0 PROGRAM LISTING

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

SVC
SVCINS=0
SVCTAG=0
.=2000

POINTER BGNSFT,BGNSW,BGNDU,BGNAU

BGNMOD MDHEDR

HEADER CVRLA,C,0,30.,0,340
.ASCII /C/
.ASCII /V/
.ASCII /R/
.ASCII /L/
.ASCII /A/
.BYTE 0
.BYTE 0
.BYTE 0
.ASCII /C/
.ASCII /O/
.WORD 0
.WORD 30.
.WORD L\$HARD
.WORD L\$SOFT
.WORD L\$HW
.WORD L\$SW
.WORD L\$LAST
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD L\$DISPATCH
.WORD 340
.WORD 0
.WORD 0
.BYTE C\$REVISION
.BYTE C\$EDIT
.WORD 0
.WORD 0
.WORD 0
.WORD L\$DVTYP
.WORD 0
.WORD 0
.WORD 0
.WORD L\$AU

944	002072	014504			.WORD	LSDU
945	002074	000000			.WORD	0
946	002076	002130			.WORD	LSDESC
947	002100	104035			EMT	ESLOAD
948	002102	000000			.WORD	0
949	002104	013732			.WORD	LSINIT
950	002106	014442			.WORD	LSCLEAN
951	002110	014440			.WORD	LSAUTO
952	002112	014432			.WORD	LSPROT
953	002114	000000			.WORD	0
954	002116	000000			.WORD	0
955	002120	000000			.WORD	0
956						
957	002122				ENDMOD	
958						
959						
960						
961						
962	002122				DEVTYPE	<RLV11>
963	002122	046122	030526	000061	.ASCII	/RLV11/
964					.EVEN	
965	002130				DESCRIPT	<CVRLACRLV11 RL01 DSKLESS DIAGNOSTIC>
966	002130	053103	046122	041501	.ASCII	/CVRLACRLV11 RL01 DSKLESS DIAGNOSTIC/
967	002136	046122	030526	020061		
968	002144	046122	030460	042040		
969	002152	045523	042514	051523		
970	002160	042040	040511	047107		
971	002166	051517	044524	000103		
972					.EVEN	
973						
974					.SBTTL	GLOBAL EQUATES
975	002174				BGNMOD	GLBEQAT
976						
977	002174				EQUALS	
978						
979					: BIT DEFINITIONS	
980					:	
981		100000			BIT15==	100000
982		040000			BIT14==	40000
983		020000			BIT13==	20000
984		010000			BIT12==	10000
985		004000			BIT11==	4000
986		002000			BIT10==	2000
987		001000			BIT09==	1000
988		000400			BIT08==	400
989		000200			BIT07==	200
990		000100			BIT06==	100
991		000040			BIT05==	40
992		000020			BIT04==	20
993		000010			BIT03==	10
994		000004			BIT02==	4
995		000002			BIT01==	2
996		000001			BIT00==	1
997					:	
998		001000			BIT9==	BIT09
999		000400			BIT8==	BIT08

1000	000200	BIT7==	BIT07	
1001	000100	BIT6==	BIT06	
1002	000040	BIT5==	BIT05	
1003	000020	BIT4==	BIT04	
1004	000010	BIT3==	BIT03	
1005	000004	BIT2==	BIT02	
1006	000002	BIT1==	BIT01	
1007	000001	BIT0==	BIT00	
1008		:		
1009		:	EVENT FLAG DEFINITIONS	
1010		:	EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION	
1011		:		
1012	000040	EF.START==	32.	; START COMMAND WAS ISSUED
1013	000037	EF.RESTART==	31.	; RESTART COMMAND WAS ISSUED
1014	000036	EF.CONTINUE==	30.	; CONTINUE COMMAND WAS ISSUED
1015	000035	EF.NEW==	29.	; A NEW PASS HAS BEEN STARTED
1016	000034	EF.PWR==	28.	; A POWER-FAIL/POWER-UP OCCURRED
1017		:		
1018		:		
1019		:	PRIORITY LEVEL DEFINITIONS	
1020		:		
1021	000340	PRI07==	340	
1022	000300	PRI06==	300	
1023	000240	PRI05==	240	
1024	000200	PRI04==	200	
1025	000140	PRI03==	140	
1026	000100	PRI02==	100	
1027	000040	PRI01==	40	
1028	000000	PRI00==	0	
1029		:		
1030		:	OPERATOR FLAG BITS	
1031		:		
1032	000004	EVL==	4	
1033	000010	LOT==	10	
1034	000020	ADR==	20	
1035	000040	IDU==	40	
1036	000100	ISR==	100	
1037	000200	UAM==	200	
1038	000400	BOE==	400	
1039	001000	PNT==	1000	
1040	002000	PRI==	2000	
1041	004000	IXE==	4000	
1042	010000	IBE==	10000	
1043	020000	IER==	20000	
1044	040000	LOE==	40000	
1045	100000	HOE==	100000	
1046	000001	DRDY=BIT0		; DRIVE READY (RLCS)
1047	000100	INTEN=BIT6		; INTERRUPT ENABLE (RLCS)
1048	100000	ERR=BIT15		; RL11 ERROR (RLCS)
1049	040000	DERR=BIT14		; RL01 DRIVE ERROR (RLCS)
1050	002000	OPI=BIT10		; OPERATION INCOMPLETE (RLCS)
1051	000200	CRDY=BIT7		; CONTROLLER READY (RLCS)
1052	000040	BA17=BIT5		; EXTENDED ADDRESS BIT 17 (RLCS)
1053	000020	BA16=BIT4		; EXTENDED ADDRESS BIT 16 (RLCS)
1054	020000	NXM=BIT13		; NON-EXISTANT MEMORY (RLCS)
1055	000000	DS0=0		; DRIVE SELECT 0 (RLCS)

.MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 20
C/RLAC.P11 31-AUG-82 11:25 GLOBAL EQUATES

SEQ 0020

```

1056      000400      DS1=BIT8      ;DRIVE SELECT 1 (RLCS)
1057      001000      DS2=BIT9      ;DRIVE SELECT 2 (RLCS)
1058      001400      DS3=BIT8!BIT9 ;DRIVE SELECT 3 (RLCS)
1059      000000      MAINT=0        ;MAINTENANCE FUNCTION-RLV11
1060      000002      WRCHK=BIT1     ;WRITE CHECK FUNCTION
1061      000004      GSTAT=BIT2     ;GET STATUS FUNCTION
1062      000006      SEEK=BIT2!BIT1 ;SEEK FUNCTION
1063      000010      RDHDR=BIT3     ;READ HEADER FUNCTION
1064      000012      WRITE=BIT3!BIT1 ;WRITE DATA FUNCTION
1065      000014      READ=BIT3!BIT2  ;READ DATA FUNCTION
1066      000202      GODRVR=BIT1!BIT7 ;CRDY AND DRDY
1067      000010      DRST=BIT3      ;DRIVE RESET (RLDA)
1068      000002      GSBIT=BIT1     ;GET STATUS BIT (RLDA)
1069      000001      MK=BIT0        ;MARKER BIT (RLDA)
1070      000004      SIGN=BIT2      ;SIGN BIT (RLDA)
1071      000100      RHHS=BIT6      ;HEAD SELECT IN READ HEADER
1072      000100      STHS=BIT6      ;HEAD SELECT IN STATUS BACK
1073      000020      DAHS=BIT4      ;HEAD SELECT IN SEEK
1074
1075      ;OFFSET FOR HARDWARE P-TABLE
1076
1077      000000      CSR=0
1078      000002      VECT=2
1079      000004      PRIOR=4
1080      000006      DRBT=6
1081      000010      CNT=10
1082      000012      LTYPE=12
1083
1084      ;OFFSET FOR SOFTWARE P-TABLE
1085
1086      000000      DLT=0
1087      000002      ELT=2
1088      000004      SIZE=4
1089
1090      002174      ENDMOD
1091
1092      .SBTTL      GLOBAL DATA
1093
1094      002174      BGNMOD      GLBDAT
1095
1096      .SBTTL      GLOBAL DATA
1097
1098      002174      000000      UUT:      .WORD      0
1099      002176      000000      UNITST: .WORD      0
1100      002200      000000      RLCS:    .WORD      0
1101      002202      000000      RLBA:    .WORD      0
1102      002204      000000      RLDA:    .WORD      0
1103      002206      000000      RLMP:    .WORD      0
1104      002210      000000      BCSR:    .WORD      0
1105      002212      000000      BPRIOR:   .WORD      0
1106      002214      000000      BVEC:    .WORD      0
1107      002216      000000      DRIVE:   .WORD      0
1108      002220      000000      B.CS:    .WORD      0
1109      002222      000000      B.BA:    .WORD      0
1110      002224      000000      B.DA:    .WORD      0
1111      002226      000000      B.MP:    .WORD      0

```

;DRIVE UNDER TEST

1112	002230	000000	DERFLG: .WORD		
1113	002232	000000	E.CS: .WORD	0	
1114	002234	000000	E.BA: .WORD	0	
1115	002236	000000	E.DA: .WORD	0	
1116	002240	000000	E.MP: .WORD	0	
1117	002242	000000	E.MP1: .WORD	0	
1118	002244	000000	PFLG: .WORD	0	;PROCESSOR TYPE 0=UNIBUS 1=Q-BUS
1119	002246	000000	TRPFLG: .WORD	0	
1120	002250	000000	INTFLG: .WORD	0	;INTERRUPT OCCURANCE FLAG
1121	002252	000000	LDCSR: .WORD	0	;LOCATION TO FORM RLCS
1122	002254	120001	XPOLY: .WORD	120001	
1123	002256	000004	ERRVEC: .WORD	4	
1124	002260	000000	BCCFBK: .WORD	0	;LOCATION USED BY 'SIMBCC'
1125	002262	000000	CALBCC: .WORD	0	;LOCATION USED BY 'SIMBCC'
1126	002264	000000	TEMP2: .WORD	0	;LOCATION USED BY 'SIMBCC'
1127	002266	000000	TEMP3: .WORD	0	;LOCATION USED BY 'SIMBCC'
1128	002270	000000	TEMP4: .WORD	0	;LOCATION USED BY 'SIMBCC'
1129	002272	000000	TEMP5: .WORD	0	
1130	002274	000000	TEMP1: .WORD	0	
1131	002276	000000	TMP0: .WORD	0	
1132	002300	000000	TMP1: .WORD	0	
1133	002302	000000	TMP2: .WORD	0	
1134	002304	000000	CHECK: .WORD	0	
1135	002306	000000	GDDAT: .WORD	0	
1136	002310	000000	BDDAT: .WORD	0	
1137	002312	000000	GCRCP: .WORD	0	
1138	002314	000000	GDCRCA: .WORD	0	
1139	002316	000000	GDCRCB: .WORD	0	
1140	002320	000000	GDDATP: .WORD	0	
1141	002322	000000	GDDATM: .WORD	0	
1142	002324	000000	MATFLG: .WORD	0	
1143	002326	000000	ERRLMT: .WORD	0	
1144	002330	000000	WHY: .WORD	0	;REASON FOR DROP IN AUTOSIZE
1145	002332	000000	T.CNTRL: .WORD	0	
1146	002334	000000	IMPFNC: .WORD	0	
1147	002336	000233	OPIMN: .WORD	155.	
1148	002340	001212	OPIMX: .WORD	650.	
1149	002342	176543	HINUM: .WORD	176543	
1150	002344	123456	LONUM: .WORD	123456	
1151	002346	000000	TEMLO: .WORD	0	
1152	002350	000000	TEMHI: .WORD	0	
1153	002352	000000	PATSAV: .WORD	0	
1154	002354	000000	DELCNT: .WORD	0	
1155	002356	000000	LFLG: .WORD	0	
1156	002360	000000	SAVCNT: .WORD	0	
1157	002362	000000	ERPOINT: .WORD	0	
1158	002364	000100	ERCOUNT: .BLKW	64.	;ERROR COUNTER FOR ALL UNITS
1159					
1160			.SBTTL	PATTERNS FOR REGISTER R/W	
1161			:		
1162			:	PATTERNS USED FOR LOADING/READING REGISTERS	
1163					
1164	002564	000000	BEGPAT: 0		;GROWING 1
1165	002566	000001	1		
1166	002570	000003	3		
1167	002572	000007	7		

1168	002574	000017	17
1169	002576	000037	37
1170	002600	000077	77
1171	002602	000177	177
1172	002604	000377	377
1173	002606	000777	777
1174	002610	001777	1777
1175	002612	003777	3777
1176	002614	007777	7777
1177	002616	017777	17777
1178	002620	037777	37777
1179	002622	077777	77777
1180	002624	177777	177777
1181	002626	177776	177776
1182	002630	177774	177774
1183	002632	177770	177770
1184	002634	177760	177760
1185	002636	177740	177740
1186	002640	177700	177700
1187	002642	177600	177600
1188	002644	177400	177400
1189	002646	177000	177000
1190	002650	176000	176000
1191	002652	174000	174000
1192	002654	170000	170000
1193	002656	160000	160000
1194	002660	140000	140000
1195	002662	100000	100000
1196			
1197	002664	000000	000000
1198	002666	000001	1
1199	002670	000002	2
1200	002672	000004	4
1201	002674	000010	10
1202	002676	000020	20
1203	002700	000040	40
1204	002702	000100	100
1205	002704	000200	200
1206	002706	000400	400
1207	002710	001000	1000
1208	002712	002000	2000
1209	002714	004000	4000
1210	002716	010000	10000
1211	002720	020000	20000
1212	002722	040000	40000
1213	002724	100000	100000
1214	002726	177777	177777
1215	002730	177776	177776
1216	002732	177775	177775
1217	002734	177773	177773
1218	002736	177767	177767
1219	002740	177757	177757
1220	002742	177737	177737
1221	002744	177677	177677
1222	002746	177577	177577
1223	002750	177377	177377

;GROWING 0

;WALKING 1

;WALKING 0

1224	002752	176777	176777
1225	002754	175777	175777
1226	002756	173777	173777
1227	002760	167777	167777
1228	002762	157777	157777
1229	002764	137777	137777
1230	002766	077777	077777
1231	002770	177777	177777
1232	002772	000000	000000

ENDPAT: 000000

.SBTTL PATTERNS FOR MAINT. CRC TEST OF SERIAL DATA PATH
 PATCRC: 155552

1233			133330
1234			066663
1235			125247
1236	002774	155552	052522
1237	002776	133330	177774
1238	003000	066663	000374
1239	003002	125247	022217
1240	003004	052522	044441
1241	003006	177774	166663
1242	003010	000374	144441
1243	003012	022217	033330
1244	003014	044441	011106
1245	003016	166663	070704
1246	003020	144441	107065
1247	003022	033330	111106
1248	003024	011106	167353
1249	003026	070704	156732
1250	003030	107065	146311
1251	003032	111106	135670
1252	003034	167353	114626
1253	003036	156732	104205
1254	003040	146311	073564
1255	003042	135670	063143
1256	003044	114626	042101
1257	003046	104205	031460
1258	003050	073564	021037
1259	003052	063143	010416
1260	003054	042101	000000
1261	003056	031460	
1262	003060	021037	
1263	003062	010416	
1264	003064	000000	

CRCEND: 000000

.DATA PATTERNS FOR MAINTENANCE TEST
 PATDAT: 155555

1265			133333
1266			066666
1267	003066	155555	125252
1268	003070	133333	052525
1269	003072	066666	177777
1270	003074	125252	000000
1271	003076	052525	107070
1272	003100	177777	070707
1273	003102	000000	144444
1274	003104	107070	033333
1275	003106	070707	011111
1276	003110	144444	022222
1277	003112	033333	
1278	003114	011111	
1279	003116	022222	

.MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 24
 CVRLAC.P11 31-AUG-82 11:25 PATTERNS FOR MAINT. CRC TEST OF SERIAL DATA PATH

SEQ 0024

1280	003120	044444	044444
1281	003122	111111	111111
1282	003124	166666	166666
1283	003126	010421	010421
1284	003130	021042	021042
1285	003132	031463	031463
1286	003134	042104	042104
1287	003136	063146	063146
1288	003140	073567	073567
1289	003142	104210	104210
1290	003144	114631	114631
1291	003146	135673	135673
1292	003150	146314	146314
1293	003152	156735	156735
1294	003154	167356	167356
1295	003156	000000	000000

ENDDAT: 000000

;PATTERNS FOR TEST OF RLCS

1296		
1297		
1298		
1299		
1300	003160	000000
1301	003162	000002
1302	003164	000004
1303	003166	000010
1304	003170	000020
1305	003172	000040
1306	003174	000100
1307	003176	000400
1308	003200	001000
1309	003202	001576
1310	003204	001574
1311	003206	001570
1312	003210	001560
1313	003212	001540
1314	003214	001500
1315	003216	001400
1316	003220	001576
1317	003222	001574
1318	003224	001566
1319	003226	001556
1320	003230	001536
1321	003232	001436
1322	003234	001136
1323	003236	000076
1324	003240	000006
1325	003242	000016
1326	003244	000036
1327	003246	000076
1328	003250	000176
1329	003252	000576
1330	003254	001576
1331	003256	000000
1332	003260	000240
1333		
1334		
1335	003760	000400

CSPAT: .WORD 0 ;SHIFTING 1

.WORD BIT1

.WORD BIT2

.WORD BIT3

.WORD BIT4

.WORD BIT5

.WORD BIT6

.WORD BIT8

.WORD BIT9

.WORD 1576

;GROWING 0

.WORD 1574

.WORD 1570

.WORD 1560

.WORD 1540

.WORD 1500

.WORD 1400

.WORD 1576

;SHIFT 0

.WORD 1574

.WORD 1566

.WORD 1556

.WORD 1536

.WORD 1436

.WORD 1136

.WORD 76

;GROWING 1

.WORD 6

.WORD 16

.WORD 36

.WORD 76

.WORD 176

.WORD 576

.WORD 1576

CSEND: .WORD 0

HDRBUF: .BLKW 160.

.SBTTL BUFFERS FOR RL11 MAINTENANCE FUNCTION
 BUF1: .BLKW 256.

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 25
 CVRLAC.P11 31-AUG-82 11:25 BUFFERS FOR RLV11 MAINTENANCE FUNCTION

SEQ 0025

1336	004760	000400		BU2:	.BLKW	256.
1337	005760			ENDMOD		
1338						
1339	005760			BGNMOD	GLBXT	
1340				.SBTTL	GLOBAL	TEXT
1341						
1342	005760	047516	041440	047117	NORES:	.ASCIZ /NO CONTROLLER/
	005776	047516	042040	044522	NODRY:	.ASCIZ /NO DRIVE CONNECTED/
	006021	040	051104	000126	DEMES:	.ASCIZ / DRV/
	006026	047040	046530	000	NXMMES:	.ASCIZ / NXM/
	006033	040	050117	000111	OPIES:	.ASCIZ / OPI/
	006040	044040	051103	000103	HCRCMES:	.ASCIZ / HCRC/
	006046	044040	043116	000	HNFMES:	.ASCIZ / HNF/
	006053	040	041504	000113	DCKMES:	.ASCIZ / DCK/
	006060	042040	052114	000	DLTMES:	.ASCIZ / DLT/
	006065	105	050130	042047	EXPMES:	.ASCIZ /EXP'D: COMP HNF OPI REC'D: /
	006122	047516	042440	050130	NONMES:	.ASCIZ /NO EXPECTED ERRORS FOUND/
	006153	015	000012		MSCRLF:	.ASCIZ <15><12>
	006156	000015			LF:	.ASCIZ <15>
	006160	041440	046517	000120	COMP:	.ASCIZ / COMP/
	006166	047506	041522	042105	OPIERR:	.ASCIZ /FORCED OPI(GET STATUS) CAUSED OTHER ERRORS/
	006241	116	047517	020120	NOPMES:	.ASCIZ /NOOP OPERATION-FLAG MODE/
	006272	047516	050117	047440	NOPINT:	.ASCIZ /NOOP OPERATION-INTR. MODE/
	006324	040515	047111	042524	MATMES:	.ASCIZ /MAINTENANCE OPERATION-FLAG MODE/
	006364	040515	047111	042524	MATINT:	.ASCIZ /MAINTENANCE OPERATION-INTERRUPT MODE/
	006431	103	035123	000040	ARLCS:	.ASCIZ /CS: /
	006436	041040	035101	000040	ARLBA:	.ASCIZ / BA: /
	006444	042040	035101	000040	ARLDA:	.ASCIZ / DA: /
	006452	046440	035120	000040	ARLMP:	.ASCIZ / MP: /
	006460	042502	047506	042522	BEREG:	.ASCIZ /BEFORE COMMAND: /
	006501	124	046511	020105	AFREG:	.ASCIZ /TIME OF ERROR: /
	006522	047503	052116	047522	CRTIM:	.ASCIZ /CONTROLLER TIMED OUT/
	006547	104	044522	042526	DRTIM:	.ASCIZ /DRIVE READY TIMED OUT/
	006575	103	047101	047040	EM1:	.ASCIZ /CAN NOT ADDRESS RLCS/
	006622	040503	020116	047516	EM2:	.ASCIZ /CAN NOT ADDRESS RLBA/
	006647	103	047101	047040	EM3:	.ASCIZ /CAN NOT ADDRESS RLDA/
	006674	040503	020116	047516	EM4:	.ASCIZ /CAN NOT ADDRESS RLMP/
	006721	122	041514	020123	EM5:	.ASCIZ %RLCS READ/WRITE ERROR (BIT 0 DON'T CARE)%
	006772	046122	040502	051040	EM6:	.ASCIZ %RLBA READ/WRITE ERROR%
	007020	046122	040504	051040	EM7:	.ASCIZ %RLDA READ/WRITE ERROR%
	007046	046122	040502	042440	EM10:	.ASCIZ /RLBA ERROR AFTER MAINT. FUNCTION/
	007107	117	044520	053440	EM11:	.ASCIZ /OPI WOULD NOT GENERATE INTERRUPT/
	007150	046122	040504	042440	EM12:	.ASCIZ /RLDA ERROR AFTER MAINT. FUNCTION/
	007211	116	020117	047111	EM13:	.ASCIZ /NO INTERRUPT FROM NOOP(0)/
	007243	116	047517	024120	EM14:	.ASCIZ /NOOP(0) MODIFIED RLMP/
	007271	116	047517	024120	EM15:	.ASCIZ /NOOP(0) MODIFIED RLBA/
	007317	116	047517	024120	EM16:	.ASCIZ /NOOP(0) MODIFIED RLDA/
	007345	111	052116	051105	EM17:	.ASCIZ /INTERRUPT PRIORITY FAILURE/
	007400	046122	050115	020072	EM20:	.ASCIZ /RLMP: CRC OF DA+3 ERROR (SERIAL DATA PATH)/
	007453	122	046514	035120	EM21:	.ASCIZ /RLMP: CRC OF CRC OF DA+4 ERROR (SERIAL DATA PATH)/
	007535	115	044501	052116	EM22:	.ASCIZ %MAINT. FILL/EMPTY FIFO DMA DATA TRANSFER COMPARE ERROR%
	007624	040515	047111	042524	EM23:	.ASCIZ /MAINTENANCE LAST WORD+1 FAILURE/
	007664	047516	044440	052116	EM24:	.ASCIZ /NO INTERRUPT FROM MAINT. FUNCTION/
	007726	040515	047111	042524	EM25:	.ASCIZ /MAINTENANCE FIFO ADDRESS ERROR/
	007765	115	044501	052116	EM26:	.ASCIZ /MAINTENANCE FIFO ADDRESS COMPLEMENT ERROR/
	010037	115	044501	052116	EM27:	.ASCIZ /MAINT. FORCED OPI ERROR, LESS THAN 510 WORDS/

010113	115	044501	052116	EM30:	.ASCIZ	/MAINT. FORCED OPI ERROR,MORE THAN 511 WORDS/
010167	117	044520	052040	EM31:	.ASCIZ	/OPI TIMING ERROR/
010210	051127	052111	047111	EM44:	.ASCIZ	/WRITING RLMP MODIFIED RLCS/
010243	127	044522	044524	EM45:	.ASCIZ	/WRITING RLMP MODIFIED RLBA/
010276	051127	052111	047111	EM46:	.ASCIZ	/WRITING RLMP MODIFIED RLDA/
010331	102	052111	051440	EM61:	.ASCIZ	/BIT SET INSTRUCTION ON RLCS YIELDED WRONG RESULT/
010412	044502	020124	046103	EM62:	.ASCIZ	/BIT CLEAR INSTRUCTION ON RLCS YIELDED WRONG RESULT/
010475	102	052111	051440	EM63:	.ASCIZ	/BIT SET INSTRUCTION ON RLBA YIELDED WRONG RESULT/
010556	044502	020124	046103	EM64:	.ASCIZ	/BIT CLEAR INSTRUCTION ON RLBA YIELDED WRONG RESULT/
010641	102	052111	051440	EM65:	.ASCIZ	/BIT SET INSTRUCTION ON RLDA YIELDED WRONG RESULT/
010722	044502	020124	046103	EM66:	.ASCIZ	/BIT CLEAR INSTRUCTION ON RLDA YIELDED WRONG RESULT/
011005	102	051525	051040	EM67:	.ASCIZ	/BUS RESET DID NOT CLEAR RLCS/
011042	052502	020123	042522	EM70:	.ASCIZ	/BUS RESET DID NOT CLEAR RLBA/
011077	102	051525	051040	EM71:	.ASCIZ	/BUS RESET DID NOT CLEAR RLDA/
011134	051127	052111	047111	EM72:	.ASCIZ	/WRITING RLCS MODIFIED RLBA/
011167	127	044522	044524	EM73:	.ASCIZ	/WRITING RLCS MODIFIED RLDA/
011222	051127	052111	047111	EM74:	.ASCIZ	/WRITING RLBA MODIFIED RLCS/
011254	051127	052111	047111	EM75:	.ASCIZ	/WRITING RLBA MODIFIED RLDA/
011306	051127	052111	047111	EM76:	.ASCIZ	/WRITING RLDA MODIFIED RLCS/
011341	127	044522	044524	EM77:	.ASCIZ	/WRITING RLDA MODIFIED RLBA/
011374	046122	051503	041440	EM101:	.ASCIZ	/RLCS CONTAINED FOLLOWING ERROR(S): /
011441	000170			EM102:	.BLKB	120.

011632

.EVEN

(0) 011632

ENDMOD

1343

.SBTTL GLOBAL ERRORS

1344

1345

1346

011632

BGNMOD GLBERR

1347

1348

011632

BGNMSG ERRO

1349

1350

011632 004737 012322

JSR PC,LINE1

1351

011636 004737 012356

JSR PC,LINE2

1352

1353

011642 004537 014514

JSR R5,CKERLT ;CHECK ERROR LIMIT

1354

011646

ENDMSG

1355

011646

L10000:

1356

011646 104423

TRAP C\$MSG

1357

1358

011650

BGNMSG ERR1

1359

1360

011650 004737 012322

JSR PC,LINE1

1361

1362

011654 004537 014514

JSR R5,CKERLT ;CHECK ERROR LIMIT

1363

011660

ENDMSG

1364

011660

L10001:

1365

011660 104423

TRAP C\$MSG

1366

011662

BGNMSG ERR2

1367

1368

011662 004737 012322

JSR PC,LINE1

1369

011666

PRINTB #FRMT4,GDDAT,BDDAT

1370

011666 013746 002310

MOV BDDAT,-(SP)

1371

011672 013746 002306

MOV GDDAT,-(SP)

1372

011676 012746 013014

MOV #FRMT4,-(SP)

1373	011702	012746	000003	MOV	#3,-(SP)	
1374	011706	010600		MOV	SP,R0	
1375	011710	104414		TRAP	C\$PNTB	
1376	011712	062706	000010	ADC	#10,SP	
1377	011716	004537	014514	JSR	R5,CKERLT	
1378	011722			ENDMSG		
1379	011722			L10002:		
1380	011722	104423		TRAP	C\$MSG	
1381						
1382						
1383	011724			BGNMSG	ERR3	
1384	011724	004737	012322	JSR	PC,LINE1	
1385	011730	004737	012356	JSR	PC,LINE2	
1386	011734			PRINTB	#FRMT99	
1387	011734	012746	013135	MOV	#FRMT99,-(SP)	
1388	011740	012746	000001	MOV	#1,-(SP)	
1389	011744	010600		MOV	SP,R0	
1390	011746	104414		TRAP	C\$PNTB	
1391	011750	062706	000004	ADD	#4,SP	
1392	011754			PRINTB	#FRMT14,E.BA,E.DA,TMPO,GDDAT,BDDAT	
1393	011754	013746	002310	MOV	BDDAT,-(SP)	
1394	011760	013746	002306	MOV	GDDAT,-(SP)	
1395	011764	013746	002276	MOV	TMPO,-(SP)	
1396	011770	013746	002236	MOV	E.DA,-(SP)	
1397	011774	013746	002234	MOV	E.BA,-(SP)	
1398	012000	012746	013477	MOV	#FRMT14,-(SP)	
1399	012004	012746	000006	MOV	#6,-(SP)	
1400	012010	010600		MOV	SP,R0	
1401	012012	104414		TRAP	C\$PNTB	
1402	012014	062706	000016	ADD	#16,SP	
1403	012020	004537	014514	JSR	R5,CKERLT	
1404	012024			ENDMSG		
1405	012024			L10003:		
1406	012024	104423		TRAP	C\$MSG	
1407						
1408						
1409						
1410	012026			BGNMSG	ERR4	
1411						
1412	012026	004737	012322	JSR	PC,LINE1	
1413	012032	004737	012356	JSR	PC,LINE2	
1414	012036			PRINTB	#FRMT4,GDDAT,BDDAT	
1415	012036	013746	002310	MOV	BDDAT,-(SP)	
1416	012042	013746	002306	MOV	GDDAT,-(SP)	
1417	012046	012746	013014	MOV	#FRMT4,-(SP)	
1418	012052	012746	000003	MOV	#3,-(SP)	
1419	012056	010600		MOV	SP,R0	
1420	012060	104414		TRAP	C\$PNTB	
1421	012062	062706	000010	ADD	#10,SP	
1422						
1423	012066	004537	014514	JSR	R5,CKERLT	;CHECK ERROR LIMIT
1424	012072			ENDMSG		
1425	012072			L10004:		
1426	012072	104423		TRAP	C\$MSG	
1427						
1428	012074			BGNMSG	ERR5	

```

1429
1430 012074 004737 012322      JSR      PC,LINE1
1431
1432 012100 004537 014514      JSR      R5,CKERLT      ;CHECK ERROR LIMIT
1433 012104      L10005:      ENDMMSG
1434 012104
1435 012104 104423      TRAP      C$MSG
1436
1437 012106      BGNMSG      ERR6
1438
1439 012106 004737 012322      JSR      PC,LINE1
1440 012112 004737 012574      JSR      PC,LINE3
1441 012116 004737 012356      JSR      PC,LINE2
1442
1443
1444 012122      1$:      PRINTB  #FRMT99
1445 012122 012746 013135      MOV      #FRMT99,-(SP)
1446 012126 012746 000001      MOV      #1,-(SP)
1447 012132 010600      MOV      SP,R0
1448 012134 104414      TRAP      C$PNTB
1449 012136 062706 000004      ADD      #4,SP
1450 012142 004537 014514      JSR      R5,CKERLT      ;CHECK ERROR LIMIT
1451 012146      L10006:      ENDMMSG
1452 012146
1453 012146 104423      TRAP      C$MSG
1454
1455 012150      BGNMSG      ERR7
1456
1457 012150 004737 012322      JSR      PC,LINE1
1458 012154      PRINTB  #FRMT6,BDDAT
1459 012154 013746 002310      MOV      BDDAT,-(SP)
1460 012160 012746 013211      MOV      #FRMT6,-(SP)
1461 012164 012746 000002      MOV      #2,-(SP)
1462 012170 010600      MOV      SP,R0
1463 012172 104414      TRAP      C$PNTB
1464 012174 062706 000006      ADD      #6,SP
1465
1466 012200 004537 014514      JSR      R5,CKERLT
1467
1468 012204      ENDMMSG
1469 012204      L10007:
1470 012204 104423      TRAP      C$MSG
1471
1472 012206      BGNMSG      ERR10
1473 012206 004737 012322      JSR      PC,LINE1
1474 012212 004737 012356      JSR      PC,LINE2
1475 012216 004737 012646      JSR      PC,LINE4
1476 012222      PRINTB  #FRMT99
1477 012222 012746 013135      MOV      #FRMT99,-(SP)
1478 012226 012746 000001      MOV      #1,-(SP)
1479 012232 010600      MOV      SP,R0
1480 012234 104414      TRAP      C$PNTB
1481 012236 062706 000004      ADD      #4,SP
1482 012242 004537 014514      JSR      R5,CKERLT
1483 012246      L10010:      ENDMMSG
1484 012246
  
```

1485	012246	104423		TRAP	C\$MSG
1486					
1487	012250			BGNMSG	ERR11
1488	012253	004737	012322	JSR	PC,LINE1
1489	012254	004737	012356	JSR	PC,LINE2
1490	012260			PRINTB	#FRMT10,OPIMN,OPIMX,BDDAT
1491	012260	013746	002310	MOV	BDDAT,-(SP)
1492	012264	013746	002340	MOV	OPIMX,-(SP)
1493	012270	013746	002336	MOV	OPIMN,-(SP)
1494	012274	012746	013246	MOV	#FRMT10,-(SP)
1495	012300	012746	000004	MOV	#4,-(SP)
1496	012304	010600		MOV	SP,R0
1497	012306	104414		TRAP	C\$PNTB
1498	012310	062706	000012	ADD	#12,SP
1499	012314	004537	014514	JSR	R5,CKERLT
1500	012320			ENDMSG	
1501	012320			L10011:	
1502	012320	104423		TRAP	C\$MSG
1503					
1504					
1505	012322			LINE1:	PRINTB #FRMT1,RLCS,<B,DRIVE+1>
1506	012322	005046		CLR	-(SP)
1507	012324	153716	002217	BISB	DRIVE+1,(SP)
1508	012330	013746	002200	MOV	RLCS,-(SP)
1509	012334	012746	012674	MOV	#FRMT1,-(SP)
1510	012340	012746	000003	MOV	#3,-(SP)
1511	012344	010600		MOV	SP,R0
1512	012346	104414		TRAP	C\$PNTB
1513	012350	062706	000010	ADD	#10,SP
1514	012354	000207		RTS	PC
1515					
1516	012356			LINE2:	PRINTB #FRMT2,#BEREG,#ARLCS,B.CS,#ARLBA,B.BA
1517	012356	013746	002222	MOV	B.BA,-(SP)
1518	012362	012746	006431	MOV	#ARLBA,-(SP)
1519	012366	013746	002220	MOV	B.CS,-(SP)
1520	012372	012746	006431	MOV	#ARLCS,-(SP)
1521	012376	012746	006460	MOV	#BEREG,-(SP)
1522	012402	012746	012734	MOV	#FRMT2,-(SP)
1523	012406	012746	000006	MOV	#6,-(SP)
1524	012412	010600		MOV	SP,R0
1525	012414	104414		TRAP	C\$PNTB
1526	012416	062706	000016	ADD	#16,SP
1527	012422			PRINTB	#FRMT2A,#ARLDA,B.DA,#ARLMP,B.MP
1528	012422	013746	002226	MOV	B.MP,-(SP)
1529	012426	012746	006452	MOV	#ARLMP,-(SP)
1530	012432	013746	002224	MOV	B.DA,-(SP)
1531	012436	012746	006444	MOV	#ARLDA,-(SP)
1532	012442	012746	012753	MOV	#FRMT2A,-(SP)
1533	012446	012746	000005	MOV	#5,-(SP)
1534	012452	010600		MOV	SP,R0
1535	012454	104414		TRAP	C\$PNTB
1536	012456	062706	000014	ADD	#14,SP
1537	012462			PRINTB	#FRMT2,#AFREG,#ARLCS,E.CS,#ARLBA,E.BA
1538	012462	013746	002234	MOV	E.BA,-(SP)
1539	012466	012746	006436	MOV	#ARLBA,-(SP)
1540	012472	013746	002232	MOV	E.CS,-(SP)

1541	012476	012746	006431	MOV	#ARLCS,-(SP)
1542	012502	012746	006501	MOV	#AFREG,-(SP)
1543	012506	012746	012734	MOV	#FRMT2,-(SP)
1544	012512	012746	000006	MOV	#6,-(SP)
1545	012516	010600		MOV	SP,R0
1546	012520	104414		TRAP	C\$PNTB
1547	012522	062706	000016	ADD	#16,SP
1548	012526			PRINTB	#FRMT2B,#ARLDA,E.DA,#ARLMP,E.MP,E.MP1
1549	012526	013746	002242	MOV	E.MP1,-(SP)
1550	012532	013746	002240	MOV	E.MP,-(SP)
1551	012536	012746	006452	MOV	#ARLMP,-(SP)
1552	012542	013746	002236	MOV	E.DA,-(SP)
1553	012546	012746	006444	MOV	#ARLDA,-(SP)
1554	012552	012746	012766	MOV	#FRMT2B,-(SP)
1555	012556	012746	000006	MOV	#6,-(SP)
1556	012562	010600		MOV	SP,R0
1557	012564	104414		TRAP	C\$PNTB
1558	012566	062706	000016	ADD	#16,SP
1559	012572	000207		RTS	PC

1560					
1561	012574			LINE3:	PRINTB #FRMT3,#EM101
1562	012574	012746	011374	MOV	#EM101,-(SP)
1563	012600	012746	013007	MOV	#FRMT3,-(SP)
1564	012604	012746	000002	MOV	#2,-(SP)
1565	012610	010600		MOV	SP,R0
1566	012612	104414		TRAP	C\$PNTB
1567	012614	062706	000006	ADD	#6,SP
1568	012620			PRINTB	#FRMT3,#EM102
1569	012620	012746	011441	MOV	#EM102,-(SP)
1570	012624	012746	013007	MOV	#FRMT3,-(SP)
1571	012630	012746	000002	MOV	#2,-(SP)
1572	012634	010600		MOV	SP,R0
1573	012636	104414		TRAP	C\$PNTB
1574	012640	062706	000006	ADD	#6,SP
1575	012644	000207		RTS	PC

1576					
1577	012646			LINE4:	PRINTB #FRMT3,#EM102
1578	012646	012746	011441	MOV	#EM102,-(SP)
1579	012652	012746	013007	MOV	#FRMT3,-(SP)
1580	012656	012746	000002	MOV	#2,-(SP)
1581	012662	010600		MOV	SP,R0
1582	012664	104414		TRAP	C\$PNTB
1583	012666	062706	000006	ADD	#6,SP
1584	012672	000207		RTS	PC

1585					
1586					
	012674	040445	047503	052116	FRMT1: .ASCIZ /%ACONTROLLER: %06%A DRIVE: %01/
	012734	047045	052045	052045	FRMT2: .ASCIZ /%N%T%T%06%T%06/
	012753	045	022524	033117	FRMT2A: .ASCIZ /%T%06%T%06/
	012766	052045	047445	022466	FRMT2B: .ASCIZ /%T%06%T%06%A %06/
	013007	045	022516	000124	FRMT3: .ASCIZ /%N%T/
	013014	047045	040445	054105	FRMT4: .ASCIZ /%N%AEXP'D: %06%A REC'D: %06%N/
	013052	047045	042045	022463	FRMT98: .ASCIZ /%N%D3%A WORDS BAD OUT OF 255 WORDS TRANSFERRED%N%N/
	013135	045	000116		FRMT99: .ASCIZ /%N/
	013140	047045	040445	040514	FRMT5: .ASCIZ /%N%ALAST: %06%A PRES: %06%A EXP'D: %06%N/
	013211	045	022516	040501	FRMT6: .ASCIZ /%N%AAT PROCESSOR LEVEL %06%N/

8

013246 047045 040445 040522 FRMT10: .ASCIZ /%N%ARANGE %D3%A - %D3%A MILLISECONDS WAS %D6%N/
013326 040445 051105 047522 FRMT11: .ASCIZ /%AERROR LIMIT EXCEEDED-DROPPED%N/
013367 045 022516 042101 FRMT12: .ASCIZ /%N%ADRIE DID NOT RECOVER FROM POWER FAILURE%N/
013446 047045 052045 040445 FRMT13: .ASCIZ /%N%T%A - WILL NOT TEST%N/
013477 045 041101 035101 FRMT14: .ASCIZ /%ABA: %06%A DA: %06%A ADDR: %06%A EXP'D: %06%A REC'D %06%N/

.EVEN

1587
1588
1589 013572
1590
1591 013572
1592
1593 013572
1594 013572 000005
1595 013574 174400
1596 013576 000160
1597 013600 000240
1598 013602 000000
1599 013604 000001
1600
1601 013606
1602 013606
1603
1604 013606
1605
1606 013606
1607
1608 013606
1609 013606 000003
1610
1611 013610 000000
1612 013612 000012
1613 013614 000000
1614
1615 013616
1616 013616
1617
1618 013616
1619
1620 013616
1621
1622 013616
1623 013616 000045
1624 013620 017002
1625 013622 017100
1626 013624 017176
1627 013626 017274
1628 013630 017372
1629 013632 017466
1630 013634 017562
1631 013636 017656
1632 013640 017752

ENDMOD
BGNMOD HPTCODE
BGNHW
 .WORD L10012-L\$HW/2
 .WORD 174400 ;CSR
 .WORD 160 ;VECTOR
 .WORD 240 ;PRIORITY
 .WORD 0 ;DRIVE (BITS 8,9,10)
 .WORD 1 ;11/23 = 1, 1103L = 0
ENDHW
L10012:
ENDMOD
BGNMOD SPTCODE
BGNSW
 .WORD L10013-L\$SW/2
DROP: .WORD 0
MERLMT: .WORD 10.
T.SIZE: .WORD 0
ENDSW
L10013:
ENDMOD
BGNMOD DSPCODE
DISPATCH
 37
 .WORD 37
 .WORD 11
 .WORD 12
 .WORD 13
 .WORD 14
 .WORD 15
 .WORD 16
 .WORD 17
 .WORD 18
 .WORD 19

1633	013642	020062		.WORD	T10	
1634	013644	020134		.WORD	T11	
1635	013646	020172		.WORD	T12	
1636	013650	020312		.WORD	T13	
1637	013652	020414		.WORD	T14	
1638	013654	020502		.WORD	T15	
1639	013656	020626		.WORD	T16	
1640	013660	020752		.WORD	T17	
1641	013662	021056		.WORD	T18	
1642	013664	021156		.WORD	T19	
1643	013666	021246		.WORD	T20	
1644	013670	021346		.WORD	T21	
1645	013672	021456		.WORD	T22	
1646	013674	021530		.WORD	T23	
1647	013676	021566		.WORD	T24	
1648	013700	021712		.WORD	T25	
1649	013702	022052		.WORD	T26	
1650	013704	022212		.WORD	T27	
1651	013706	022416		.WORD	T28	
1652	013710	022526		.WORD	T29	
1653	013712	022636		.WORD	T30	
1654	013714	022772		.WORD	T31	
1655	013716	023324		.WORD	T32	
1656	013720	024136		.WORD	T33	
1657	013722	025014		.WORD	T34	
1658	013724	025362		.WORD	T35	
1659	013726	025734		.WORD	T36	
1660	013730	026612		.WORD	T37	
1661						
1662	013732			ENDMOD		
1663						
1664				.SBTTL	INITIALIZATION CODE	
1665	013732			BGNMOD	INITCODE	
1666						
1667	013732			BGNINIT		
1668						
1669	013732			BRESET		
1670	013732	104433		TRAP	CSRESET	
1671	013734			READEF	#EF.PWR	:POWER UP????
1672	013734	012700	000034	MOV	#EF.PWR,RO	
1673	013740	104447		TRAP	CSREFG	
1674	013742			BCOMPLETE	CONT	:BRANCH
1675	013742	103510		BCS	CONT	
1676	013744			READEF	#EF.RESTART	:RESTART?
1677	013744	012700	000037	MOV	#EF.RESTART,RO	
1678	013750	104447		TRAP	CSREFG	
1679	013752			BCOMPLETE	START	
1680	013752	103411		BCS	START	
1681	013754			READEF	#EF.START	:START?
1682	013754	012700	000040	MOV	#EF.START,RO	
1683	013760	104447		TRAP	CSREFG	
1684	013762			BCOMPLETE	START	
1685	013762	103405		BCS	START	
1686	013764			READEF	#EF.NEW	:NEW PASS????
1687	013764	012700	000035	MOV	#EF.NEW,RO	
1688	013770	104447		TRAP	CSREFG	

```

1689 013772          BCOMPLETE      START1      ;YES, THEN RE INIT
1690 013772 103416    BCS          START1
1691 013774 000424    BR          CONTINUE
1692 013776 012737 176543 002342  START:  MOV      #176543, HINUM      ;RANDOM GEN. PRIME
1693 014004 012737 123456 002344      MOV      #123456, LONUM      ;RANDOM GEN. PRIME
1694 014012 012700 002364      MOV      #ERCOUNT, R0      ;SETUP TO CLEAR ERROR COUNTERS
1695 014016 012701 000100      MOV      #64, R1      ;GET A COUNT
1696 014022 005020      1$:          CLR      (R0)+      ;CLEAR A COUNTER
1697 014024 005301      DEC      R1
1698 014026 001375      BNE      1$      ;LOOP TILL COUNTERS CLEARED
1699 014030 013737 002344 002346  START1:  MOV      LONUM, TEMLO
1700 014036 013737 002342 002350      MOV      HINUM, TEMHI      ;NEW PRIMES AT END OF PASS
1701 014044 000407      BR          START2
1702
1703 014046          CONTINUE:      REDEF      #EF, CONTINUE      ;CONTINUE???
1704 014046 0.2700 000036      MOV      #EF, CONTINUE, R0
1705 014052 104447      TRAP      CSREFG
1706 014054          BCOMPLETE      CONT
1707 014054 103443      BCS      CONT
1708
1709 014056 005737 002174      NXT:      TST      UUT      ;DONE ALL UUT'S
1710 014062 001011      BNE      START3      ;NO
1711 014064 012737 177777 002176  START2:  MOV      #-1, UNITST
1712 014072 013737 002012 002174      MOV      L$UNIT, UUT
1713 014100 012737 002362 002362      MOV      #ERCOUNT-2, ERPOINT      ;INIT THE UNIT ERROR COUNTER
1714
1715 014106 005237 002176 002362  START3:  INC      UNITST
1716 014112 062737 000002      ADD      #2, ERPOINT      ;POINT TO PROPER ERROR COUNTER LOCATION
1717 014120 005337 002174      DEC      UUT
1718 014124          REST:      GPHARD  UNITST, R0
1719 014124 013700 002176      MOV      UNITST, R0
1720 014130 104442      TRAP      CS$GPHRD
1721 014132          BNCOMPLETE      NXT
1722 014132 103351      BCC      NXT
1723 014134 012037 002210      1$:      MOV      (R0)+, BCSR
1724 014140 012037 002214      MOV      (R0)+, BVEC
1725 014144 012037 002212      MOV      (R0)+, BPRIOR
1726 014150 012037 002216      MOV      (R0)+, DRIVE
1727 014154 012037 002332      MOV      (R0)+, T.CNTRL      ;GET CONTROLLER TYPE
1728 014160 012037 002356      MOV      (R0)+, LFLG      ;GET PROCESSOR
1729                                     ;LFLG=1, 11/23
1730                                     ;LFLG=0, 1103L ETC.
1731
1732 014164 013737 002346 002344  CONT:      MOV      TEMLO, LONUM      ;RESTORE RANDOM FOR NEXT UUT
1733 014172 013737 002350 002342      MOV      TEMHI, HINUM      ;RESTORE PRIME FOR NEXT UUT
1734 014200 013700 002210      MOV      BCSR, R0
1735 014204 010037 002200      MOV      R0, RLCS
1736 014210 062700 000002      ADD      #2, R0
1737 014214 010037 002202      MOV      R0, RLBA
1738 014220 062700 000002      ADD      #2, R0
1739 014224 010037 002204      MOV      R0, RLDA
1740 014230 062700 000002      ADD      #2, R0
1741 014234 010037 002206      MOV      R0, RLMP
1742 014240 005737 013614      TST      T.SIZE      ;DO WE WANT TO CHECK UNITS??
1743 014244 001450      BEQ      END      ;NO
1744

```

1745	014246	005037	002246	CLR	TRPFLG		:CLR OUT TRAP FLAG
1746	014252			SETVEC	ERRVEC, #TRPHAN, #340		:SETUP VECTOR TO CATCH NON-EXIST
1747	014252	012746	000340	MOV	#340, -(SP)		
1748	014256	012746	016636	MOV	#TRPHAN, -(SP)		
1749	014262	013746	002256	MOV	ERRVEC, -(SP)		
1750	014266	012746	000003	MOV	#3, -(SP)		
1751	014272	104437		TRAP	C\$SVEC		
1752	014274	062706	000010	ADD	#10, SP		
1753	014300	005777	165674	TST	@RLCS		:ACCESS CONTROLLER
1754	014304			CLRVEC	ERRVEC		:RELEASE VECTOR
1755	014304	013700	002256	MOV	ERRVEC, R0		
1756	014310	104436		TRAP	C\$CVEC		
1757	014312	005737	002246	TST	TRPFLG		:DID IT TRAP
1758	014316	001423		BEQ	END		
1759	014320	012737	005760	MOV	#NORES, WHY		:SETUP ERR MESS
1760	014326			PRINTB	#FRMT13, WHY		
1761	014326	013746	002330	MOV	WHY, -(SP)		
1762	014332	012746	013446	MOV	#FRMT13, -(SP)		
1763	014336	012746	000002	MOV	#2, -(SP)		
1764	014342	010600		MOV	SP, R0		
1765	014344	104414		TRAP	C\$PNTB		
1766	014346	062706	000006	ADD	#6, SP		
1767	014352	004737	012322	JSR	PC, LINE1		:GIVE DRIVE INFO
1768	014356			DODU	UNITST		:TELL SUPERVISOR TO DROP IT
1769	014356	013700	002176	MOV	UNITST, R0		
1770	014362	104451		TRAP	C\$DODU		
1771	014364	000634		BR	NXT		:TRY NEXT
1772	014366			END:	SETVEC	BVEC, #INTSRV, #340	
1773	014366	012746	000340	MOV	#340, -(SP)		
1774	014372	012746	016644	MOV	#INTSRV, -(SP)		
1775	014376	013746	002214	MOV	BVEC, -(SP)		
1776	014402	012746	000003	MOV	#3, -(SP)		
1777	014406	104437		TRAP	C\$SVEC		
1778	014410	062706	000010	ADD	#10, SP		
1779	014414	005037	002244	CLR	PFLG		:CLR PROCESSOR FLAG
1780	014420			READBUS			:Q-BUS
1781	014420	104407		TRAP	C\$RDBU		
1782	014422			BNCOMPLETE	1\$		
1783	014422	103002		BCC	1\$		
1784	014424	005237	002244	INC	PFLG		:NO, Q-BUS THEN
1785	014430			1\$:			
1786	014430			ENDINIT			
1787	014430			L10014:			
1788	014430	104411		TRAP	C\$, IT		
1789							
1790	014432			ENDMOD			
1791	014432			BGNPROT			
1792	014432	177777		.WORD	-1		:CSR OFFSET MAKE NOP
1793	014434	177777		.WORD	-1		:MASS BUS OFFSET MAKE NOP
1794	014436	177777		.WORD	-1		:DRIVE OFFSET MAKE NOP
1795	014440			ENDPROT			
1796							
1797	014440			BGNAUTO			
1798	014440			ENDAUTO			
1799	014440			L10016:			
1800	014440	104461		TRAP	C\$AUTO		

```

1801
1802
1803 014442          BGNMOD  CLNCODE
1804
1805 014442          BGNCLN
1806
1807 014442          SETPRI  #PRI00
1808 014442 012700 000000  MOV    #PRI00,RO
1809 014446 104441    TRAP    C$SPRI
1810
1811 014450 032777 000200 165522 1$:  BIT    #CRDY,@RLCS
1812 014456 001774      BEQ    1$
1813
1814 014460          SETPRI  #PRI07
1815 014460 012700 000340  MOV    #PRI07,RO
1816 014464 104441    TRAP    C$SPRI
1817 014466 042777 000100 165504    BIC    #INTEN,@RLCS
1818
1819 014474          CLRVEC  BVEC
1820 014474 013700 002214  MOV    BVEC,RO
1821 014500 104436    TRAP    C$CVEC
1822 014502          2$:
1823 014502          ENDCLN
1824 014502          L10017:
1825 014502 104412    TRAP    C$CLEAN
1826
1827 014504          ENDMOD
1828
1829
1830
1831 014504          BGNMOD  DRPCODE
1832
1833 014504          BGNDU
1834
1835 014504 000240    NOP
1836
1837 014506          ENDDU
1838 014506          L10020:
1839 014506 104453    TRAP    C$DU
1840
1841 014510          ENDMOD
1842
1843 014510          BGNMOD  ADDCODE
1844
1845 014510          BGNAU
1846
1847 014510 000240    NOP
1848
1849 014512          ENDAU
1850 014512          L10021:
1851 014512 104452    TRAP    C$AU
1852
1853 014514          ENDMOD
1854
1855
1856          .SBTTL  GLOBAL SUBROUTINES
  
```

```

1857
1858 014514 BGNMOD GLBSUB
1859
1860 014514 CKERLT: INLOOP
1861 014514 104420 TRAP C$INLP
1862 014516 BCOMPLETE 99$
1863 014516 103427 BCS 99$
1864 014520 005737 013610 TST DROP ;DROP ON ERROR LIMIT?
1865 014524 001424 BEQ 99$ ;NO
1866 014526 005277 165630 INC @ERPOINT ;COUNT THE UNIT ERROR DETECTED
1867 014532 027737 165624 013612 CMP @ERPOINT, MERLMT ;REACHED THE ERROR LIMIT?
1868 014540 002416 BLT 99$ ;NO
1869
1870 014542 PRINTF #FRMT11
1871 014542 012746 013326 MOV #FRMT11, -(SP)
1872 014546 012746 000001 MOV #1, -(SP)
1873 014552 010600 MOV SP, R0
1874 014554 104417 TRAP C$PRINTF
1875 014556 062706 000004 ADD #4, SP
1876 014562 004737 012322 JSR PC, LINE1
1877 014566 DODU UNITST ;DROP THE UNIT
1878 014566 013700 002176 MOV UNITST, R0
1879 014572 104451 TRAP C$DODU
1880 014574 DOCLN
1881 014574 104444 TRAP C$DCLN
1882 014576 99$:
1883 014576 000205 RTS R5
1884
1885
1886
1887 .SBTTL ROUTINE TO CHECK FOR CONTROLLER ERRORS
1888
1889 *****
1890 *THIS ROUTINE WILL CHECK RLCS FOR ERRORS AND PRINT THEM
1891 *ACCORDINGLY. IT WILL MERGE THE ERROR PRINTOUT WITH THE TEST
1892 *ERROR MESSAGE.
1893 *
1894 *EXAMPLE: RLCS CONTAINED FOLLOWING ERROR(S):
1895 *          DRV OPI MCRC HNF
1896 *          MAINTENANCE OPERATION-INTERRUPT MODE
1897 *
1898 *
1899 *
1900 *ROUTINE USES R0,R1 AND PICKS HEADER FROM R3
1901 *
1902 *          CALL JSR R5,CHERR
1903 *
1904 *
1905 *
1906
1907 014600 005037 002230 CHERR: CLR DERFLG ;CLEAR OUT DRIVE ERROR FLAG
1908 014604 032737 176000 002232 BIT #176000,E.CS ;ANY ERRORS SET
1909 014612 001001 BNE 199$ ;IF YES, INVESTIGATE
1910 014614 000205 RTS R5 ;NO, EXIT
1911 014616 023727 002334 000004 199$: CMP TMPFNC, #GSTAT ;FUNCTION-NOP, RESET, GETSTATUS
1912 014624 002401 BLT 98$ ;YES, GO CHECK IF ONLY DRIVE ERROR

```

1913	014626	000414			BR	1\$	:YES SERVICE ERROR
1914	014630	023727	002334	000002	LMP	TMPCNC,#WRCHK	
1915	014636	001410			BEQ	1\$	
1916	014640	013700	002232		MOV	E.CS,R0	:GET E.CS
1917	014644	042700	001777		BIC	#1777,R0	
1918	014650	022700	140000		CMP	#140000,R0	:DRIVE ERROR ALONE?
1919	014654	001001			BNE	1\$	:NO, GO SERVICE
1920	014656	000205			RTS	R5	:YES, EXIT
1921							
1922	014660	012701	011441		MOV	#EM102,R1	:GET START OF STRING
1923	014664	005737	002232		TST	E.CS	:IS COMPOSITE ERROR SET?(BETTER BE)
1924	014670	100003			BPL	99\$	:IT'S NOT SOMETHING IS WRONG
1925	014672	004537	015502		JSR	R5,FIX	:YES, PUT 'COMP' IN STRING
1926	014676	006160			COMP		: 'COMP'
1927	014700	032737	040000	002232	BIT	#DERR,E.CS	:DRIVE ERROR SET?
1928	014706	001405			BEQ	3\$	:NO, CONTINUE
1929	014710	005237	002230		INC	DERFLG	:SET DRV ERROR FLAG
1930	014714	004537	015502		JSR	R5,FIX	:YES, PUT 'DRV' INTO STRING
1931	014720	006021			DEMES		: 'DRV'
1932	014722	032737	020000	002232	BIT	#NXM,E.CS	:NON-EXISTENT MEMORY ERROR?
1933	014730	001403			BEQ	4\$	:NO, CONTINUE
1934	014732	004537	015502		JSR	R5,FIX	:YES, PUT 'NXM' INTO STRING
1935	014736	006026			NXMMES		: 'NXM'
1936	014740	032737	002000	002232	BIT	#OPI,E.CS	:IS OPI SET?
1937	014746	001422			BEQ	6\$	:NO, GO CHECK BITS 11 & 12
1938	014750	004537	015502		JSR	R5,FIX	:PUT 'OPI' INTO STRING
1939	014754	006033			OPIMES		: 'OPI'
1940	014756	032737	004000	002232	BIT	#BIT11,E.CS	:HEADERCRC ERROR?
1941	014764	001403			BEQ	5\$	:NO, GO CHECK HEADER NOT FOUND
1942	014766	004537	015502		JSR	R5,FIX	:GO PUT 'HCRC' IN STRING
1943	014772	006040			HCRCMES		: 'HCRC'
1944	014774	032737	010000	002232	BIT	#BIT12,E.CS	:HEADER NOT FOUND?
1945	015002	001422			BEQ	8\$	:NO, GO PUT 'CRLF' IN STRING
1946	015004	004537	015502		JSR	R5,FIX	:PUT 'HNF' IN STRING
1947	015010	006046			HNFMES		: 'HNF'
1948	015012	000416			BR	8\$	:PUT 'CRLF' IN STRING
1949	015014	032737	004000	002232	BIT	#BIT11,E.CS	:DATA CRC ERROR?
1950	015022	001403			BEQ	7\$	:NO, GO CHECK DATA LATE
1951	015024	004537	015502		JSR	R5,FIX	:PUT 'DCK' IN STRING
1952	015030	006053			DCKMES		: 'DCK'
1953	015032	032737	010000	002232	BIT	#BIT12,E.CS	:DATA LATE ERROR?
1954	015040	001403			BEQ	8\$	:NO, GO PUT IN 'CRLF'
1955	015042	004537	015502		JSR	R5,FIX	:PUT 'DLT' IN STRING
1956	015046	006060			DLTMES		: 'DLT'
1957	015050	004537	015502		JSR	R5,FIX	
1958	015054	006153			MSCRLF		
1959	015056	004537	015502		JSR	R5,FIX	
1960	015062	000000			RESTMS:	.WORD 0	:HEADER FROM TEST
1961	015064	105011			CLRB	(R1)	:PUT TERMINATOR IN
1962							
1963	015066				ERRDF	300.,LF,ERR6	
1964	015066	104455			TRAP	C\$ERDF	
1965	015070	000454			.WORD	300	
1966	015072	006156			.WORD	LF	
1967	015074	012106			.WORD	ERR6	
1968							

```

1969 015076 000205          RTS      R5          ;EXIT ROUTINE
1970
1971          ;*****
1972          ;* ROUTINE TO LOAD RLCS WITH FUNCTION TO BE PERFORMED FOR RL11
1973          ;* CALL:      JSR      R5,LDFUNC
1974          ;*          .WORD          ;BITS TO BE LOADED, FUNCTION
1975          ;*          ;AND INTR ENABLE ONLY
1976          ;*
1977          ;*
1978          ;*
1979 015100 012537 002252      LDFUNC: MOV      (R5)+,LDCSR      ;GET BITS TO LOAD
1980 015104 005737 002230          TST      DERFLG
1981 015110 001424          BEQ      98$
1982 015112 013746 002220          MOV      B.CS,-(SP)
1983 015116 012777 000013 165060      MOV      #13,@RLDA
1984 015124 012737 000004 002220      MOV      #GSTAT,B.CS
1985 015132 053737 002216 002220      BIS      DRIVE,B.CS
1986 015140 013777 002220 165032      MOV      B.CS,@RLCS
1987 015146 012637 002220          MOV      (SP)+,B.CS
1988 015152 032777 000200 165020 99$: BIT      #200,@RLCS
1989 015160 001774          BEQ      99$
1990 015162 010346          98$: MOV      R3,-(SP)      ;SAVE R3
1991 015164 042737 177661 002252      BIC      #177661,LDCSR      ;CLEAR ALL BUT FUNC & INTR EN
1992 015172 013737 002252 015316      MOV      LDCSR,FNDFNC      ;SAVE FUNCTION
1993 015200 042737 000100 015316      BIC      #INTEN,FNDFNC      ;ONLY FUNCTION
1994 015206 013737 015316 002334      MOV      FNDFNC,TMPFNC
1995 015214 012703 015320          MOV      #HDLRST,R3      ;GET HEADER LIST
1996 015220 006237 015316          ASR      FNDFNC      ;ALIGN TO RIGHT
1997 015224 001404          BEQ      2$
1998 015226 022323          1$: CMP      (R3)+,(R3)+      ;BUMP R3 BY 4
1999 015230 005337 015316          DEC      FNDFNC      ;FOUND IT
2000 015234 001374          BNE      1$      ;NO,KEEP LOOKING
2001 015236 032737 000100 002252 2$: BIT      #INTEN,LDCSR      ;YES,DO WE WANT FLAG OR INTR
2002 015244 001401          BEQ      3$      ;FLAG BRANCH
2003 015246 005727          TST      (R3)+      ;INTR POINT TO THAT ONE
2004 015250 011305          3$: MOV      (R3),R3      ;SET HEADER
2005 015252 010337 015062          MOV      R3,RESTMS      ;SET UP HEADER
2006 015256 053737 002216 002252      BIS      DRIVE,LDCSR      ;SELECT DRIVE
2007 015264 052737 000200 002252 4$: BIS      #200,LDCSR      ;CONTROLLER READY
2008 015272 013777 002252 164700      MOV      LDCSR,@RLCS
2009 015300 004537 015660          JSR      R5,BEFORE
2010 015304 042777 000200 164666 5$: BIC      #200,@RLCS
2011 015312 012603          MOV      (SP)+,R3      ;RESTORE R3
2012 015314 000205          RTS      R5      ;EXIT
2013
2014 015316 000000          FNDFNC: .WORD      0
2015
2016 015320 006241          HDRLST: NOPMES
2017 015322 006272          NOPINT
2018
2019
2020          ;*****
2021          ;* THIS ROUTINE WILL CHECK RLV11 CSR FOR COMP,HNF AND OPI ERRORS
2022          ;* IN THE MAINTENANCE FORCED OPI TESTS. IT WILL MERGE THE ERROR PRINTOUT
2023          ;* WITH THE TEST ERROR MESSAGE. DRIVE ERRORS WILL BE IGNORED.
2024          ;*

```

```

2025      :      CALL      JSR      R5,CHKOP1
2026      :
2027      :      CHKOP1: MOV      R1,-(SP)
2028      :      MOV      #112000,R1      :EXPECTED RESULTS
2029      :      CLR      MATFLG      :CLEAR ERROR FOUND FLAG
2030      :      BIC      E.CS,R1      :CHECK COMP,HNF,OPI
2031      :      TST      R1
2032      :      BNE      1$      :EXPECTED ERRORS NOT SET
2033      :      BR      6$      :ALL EXPECTED ERRORS SET,EXIT
2034      :      MOV      #EM102,R1      :GET START OF TEXT STRING
2035      :      JSR      R5,FIX      :STORE MESSAGE
2036      :      EXPMES      :EXPECTED
2037      :      BIT      #BIT15,E.CS      :IS COMP SET?
2038      :      BEQ      2$      :NO,CONTINUE ERROR SEARCH
2039      :      INC      MATFLG      :YES,SET ERROR FOUND
2040      :      JSR      R5,FIX      :STORE COMP MESSAGE
2041      :      COMP
2042      :      BIT      #BIT12,E.CS      :IS HNF SET?
2043      :      BEQ      3$      :NO,CONTINUE ERROR SEARCH
2044      :      INC      MATFLG      :YES,SET ERROR FOUND
2045      :      JSR      R5,FIX      :STORE HNF MESSAGE
2046      :      HNFMES
2047      :      BIT      #BIT10,E.CS      :IS OPI SET?
2048      :      BEQ      4$      :NO,COMPLETE MESSAGE
2049      :      INC      MATFLG      :YES,SET ERROR FOUND
2050      :      JSR      R5,FIX      :STORE OPI MESSAGE
2051      :      OPIMES
2052      :      TST      MATFLG      :CHECK IF EXPECTED ERRORS FOUND
2053      :      BNE      5$
2054      :      JSR      R5,FIX      :STORE MESSAGE
2055      :      NONMES      :NO EXPECTED ERRORS FOUND
2056      :      JSR      R5,FIX
2057      :      MSCRLF
2058      :      CLRB      (R1)      :STORE MESSAGE TERMINATOR
2059      :      TST      (R5)+      :RETURN TO PRINT ERROR
2060      :      MOV      (SP)+,R1
2061      :      RTS      R5
2062      :*****
2063      :*ROUTINE TO MOVE ASCII STRINGS
2064      :*USES REGISTERS R1 - WHERE STRING IS BEING BUILT
2065      :*
2066      :*      CALL      JSR      R5,FIX
2067      :*      .WORD      :ADDRESS OF STRING TO MOVE
2068      :*
2069      :      MOV      (R5)+,R0      :GET ADDRESS AND MOVE RETURN
2070      :      MOVB      (R0)+,(R1)+      :GET BYTE AND UPDATE
2071      :      BNE      1$      :WATCH 0 BYTE TERMINATOR
2072      :      TSTB      -(R1)      :BACK UP OVER ZERO BYTE
2073      :      RTS      R5      :EXIT
2074
2075
2076      :*****
2077      :*RLV11 MAINTENANCE SUBROUTINE FOR CRC CALCULATIONS
2078      :*ROUTINE TO RETRIEVE PATTERN AND CALCULATE CRC OF PATTERN+3
2079      :*AND CRC OF CRC OF PATTERN+4.
2080      :*CRC OF PATTERN+3 WILL BE STORED IN 'GDCRCA'.
  
```

MAIN. MACV11 30(1046) 31-AUG-82 11:50 PAGE 40
 (VRLAC.P11 31-AUG-82 11:25 ROUTINE TO CHECK FOR CONTROLLER ERRORS

SEQ 0040

```

2081      ;CRC OF CRC OF PATTERN+4 WILL BE STORED IN 'GDCRCB'.
2082      ;PATTERN WILL BE STORED IN 'GDDATA'.
2083      :
2084      :      CALL      JSR      R5,CALCRC
2085      :      .WORD      :PATTERN IN DA
2086
2087      015514 012537 002312      CALCRC: MOV      (R5)+,GCRCP1      ;STORE PATTERN
2088      015520 013737 002312 002274      MOV      GCRCP1,TEMP1
2089      015526 113737 002274 002272      MOV      TEMP1,TEMP5
2090      015534 062737 000003 002272      ADD      #3,TEMP5      ;ADD 3 TO PATTERN
2091      015542 113737 002272 002274      MOV      TEMP5,TEMP1
2092      015550 013737 002274 015564      MOV      TEMP1,1$
2093      015556 004537 016464      JSR      R5,SIMBCC      ;CALCULATE EXPECTED CRC
2094      015562 000020      1$:      16.      ;DATA BITS
2095      015564 000000      .WORD      0      ;INITIAL PATTERN+3
2096      015566 000000      .WORD      0
2097      015570 013737 002262 002314      MOV      CALBCC,GDCRCA      ;SAVE CRC OF PATTERN+3
2098      015576 005237 002272      INC      TEMP5      ;VALUE=PATTERN+4
2099      015602 113737 002272 002274      MOV      TEMP5,TEMP1
2100      015610 013737 002274 015624      MOV      TEMP1,2$
2101      015616 004537 016464      JSR      R5,SIMBCC      ;CALCULATE EXPECTED CRC
2102      015622 000020      16.      ;DATA BITS
2103      015624 000000      2$:      .WORD      0      ;INITIAL PATTERN+4
2104      015626 000000      .WORD      0      ;STARTING CRC=0
2105      015630 013737 002262 015644      MOV      CALBCC,3$      ;STORE CRC FOR NEXT CALL
2106      015636 004537 016464      JSR      R5,SIMBCC      ;CAL. CRC OF CRC OF DA+4
2107      015642 000020      16.      ;DATA BITS
2108      015644 000000      3$:      .WORD      0      ;CRC OF DA+4
2109      015646 000000      .WORD      0      ;STARTING CRC=0
2110      015650 013737 002262 002316      MOV      CALBCC,GDCRCB      ;SAVE CRC OF CRC OF DA+4
2111      015656 000205      RTS      R5
2112
2113      ;LOAD REGISTERS BEFORE FUNCTION
2114      ;CALL: JSR      R5,BEFORE
2115
2116      015660 017737 164314 002220      BEFORE: MOV      @RLCS,B.CS      ;READ CS
2117      015666 017737 164310 002222      MOV      @RLBA,B.BA      ;READ BA
2118      015674 017737 164304 002224      MOV      @RLDA,B.DA      ;READ DA
2119      015702 017737 164300 002226      MOV      @RLMP,B.MP      ;READ MP
2120      015710 000205      RTS      R5
2121
2122
2123      ;LOAD REGISTERS AT ERROR
2124      ;CALL: JSR      R5,AFTER
2125
2126      015712 017737 164262 002232      AFTER: MOV      @RLCS,E.CS      ;READ CS
2127      015720 017737 164256 002234      MOV      @RLBA,E.BA      ;READ BA
2128      015726 017737 164252 002236      MOV      @RLDA,E.DA      ;READ DA
2129      015734 017737 164246 002240      MOV      @RLMP,E.MP      ;READ MP
2130      015742 017737 164240 002242      MOV      @RLMP,E.MP1      ;READ MP
2131      015750 000205      RTS      R5
2132
2133      ;ROUTINE TO SETUP BUFFERS FOR RLV11 MAINTENANCE FUNCTION
2134      ;BUF1 IS SET WITH 256 WORDS OF PATTERN
2135      ;BUF2 IS CLEARED BEFORE MAINTENANCE FUNCTION
2136      :      CALL      JSR      R5,SETPAT

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 41
 (VRLAC.P11 31-AUG-82 11:25 ROUTINE TO CHECK FOR CONTROLLER ERRORS

SEQ 0041

```

2137      :      .WORD      :PATTERN FOR BUFFER
2138      :
2139      SETPAT: MOV      R1,-(SP)
2140      MOV      R2,-(SP)
2141      MOV      (R5)+,GDDATP
2142      MOV      #BUF1,R1      :FIRST BUFFER START
2143      MOV      #256,R2
2144      1$:      MOV      GDDATP,(R1)+
2145      DEC      R2
2146      BNE      1$      :STORE PATTERN IN 256 WORDS
2147      MOV      #BUF2,R1      :START OF SECOND BUFFER
2148      MOV      #255,R2
2149      2$:      CLR      (R1)+
2150      DEC      R2
2151      BNE      2$      :CLEAR 255 WORDS OF SECOND BUFFER
2152      MOV      #123456,(R1)+ :STORE IN LAST BUFFER WORD
2153      MOV      (SP)+,R2
2154      MOV      (SP)+,R1
2155      RTS      R5
2156
2157      :ROUTINE TO DELAY IN MSECS
2158      :LFLG = 1      DELAY MSECS FOR 11/23
2159      :LFLG = 0      DELAY MSECS FOR 1103L ETC.
2160
2161      :CALL
2162      JSR      R5,WDELAY
2163      40.      :40 MSECS
2164
2165      WDELAY: MOV      R1,-(SP)
2166      MOV      R2,-(SP)
2167      MOV      (R5)+,R2      :APPROX MSEC DELAY
2168      TST      LFLG      :CHECK PROCESSOR FLAG
2169      BNE      1$      :BRANCH IF 11/23
2170      MOV      #120.,DELCNT :LSI-11 APPROX 1 MSEC LOOP
2171      BR      2$
2172      1$:      MOV      #300.,DELCNT :11/23 APPROX 1 MSEC LOOP
2173      2$:      MOV      DELCNT,R1
2174      3$:      DEC      R1      :START LOOP
2175      BNE      3$
2176      DEC      R2      :CHECK ON MSECS REQUESTED
2177      BNE      2$      :BRANCH AND DO ANOTHER LOOP
2178      MOV      (SP)+,R2      :SETUP FOR RETURN AFTER DELAY
2179      MOV      (SP)+,R1
2180      RTS      R5
2181
2182      :ROUTINE TO LOAD RLCS WITH RLV11 MAINT. FUNCTION
2183      :EITHER FLAG DRIVEN OR INTERRUPT MODE.
2184      :      CALL      JSR      R5,LDFUN
2185      :      .WORD      :MAINT!INTEN
2186      :      .WORD      :WORD COUNT COMP.
2187      :      .WORD      :MAINTENANCE MESSAGE
2188
2189      LDFUN.  MOV      (R5)+,LDCSR      :GET FUNCTION
2190      MOV      (R5)+,@RLMP      :LOAD WORD COUNT
2191      MOV      (R5)+,RESTMS      :GET MESSAGE
2192      CLR      TMPFNC      :CLEAR FUNCTION STORAGE

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 42
 (VRLAC.P11 31-AUG-82 11:25

ROUTINE TO CHECK FOR CONTROLLER ERRORS

SEQ 0042

```

2193 016134 012777 003760 164040      MOV    #BUF1,@RLBA      ;SET BA REGISTER
2194 016142 013777 002312 164034      MOV    GCRCPT,@RLDA    ;LOAD DA REGISTER
2195 016150 042737 177661 002252      BIC    #177661,LDCSR   ;CLEAR ALL BUT FUNC.+INT.
2196 016155 053737 002216 002252      BIS    DRIVE,LDCSR    ;SELECT DRIVE
2197 016164 052737 000200 002252      BIS    #200,LDCSR     ;CONTROLLER READY
2198 016172 013777 002252 164000      MOV    LDCSR,@RLCS    ;LOAD CS REGISTER
2199 016200 004537 015660          JSR    R5,BEFORE        ;STORE REGISTERS BEFORE OPERATION
2200 016204 042777 000200 163766      BIC    #200,@RLCS     ;CLEAR CONTROLLER READY
2201 016212 000205          RTS    R5                ;RETURN

```

;ROUTINE TO SETUP COMPLEMENT BUFFERS FOR RLV11 MAINTENANCE FUNCTION

;BUF1 IS SET WITH PATTERN

;BUF1+1 IS SET WITH COMPLEMENT OF PATTERN

```

2202          CALL    JSR    R5,SETCMP
2203          .WORD
2204          .WORD      ;PATTERN FOR BUFFER

```

```

2205          SETCMP: MOV    R1,-(SP)
2206          MOV    R2,-(SP)
2207          MOV    (R5)+,GDDATP
2208          MOV    #BUF1,R1      ;FIRST BUFFER START
2209          MOV    #256.,R2      ;BUFFER COUNT
2210          MOV    GDDATP,GDATMP ;STORE DATA PATTERN FOR BUF FILL
2211          1$: MOV    GDATMP,(R1)+
2212          COM    GDATMP        ;STORE COMP. IN NEXT BUF LOCATION
2213          DEC    R2
2214          BNE    1$           ;CHECK FOR BUFFER END
2215          MOV    #BUF2,R1      ;SETUP TO CLEAR BUF2
2216          MOV    #255.,R2
2217          2$: CLR    (R1)+
2218          DEC    R2
2219          BNE    2$           ;CHECK FOR BUF2 END
2220          MOV    #123456,(R1)+ ;STORE IN LAST BUFFER WORD
2221          MOV    (SP)+,R2
2222          MOV    (SP)+,R1
2223          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

```

2224          CALL    JSR    R5,SETRAN

```

```

2225          SETRAN: MOV    R1,-(SP)
2226          MOV    R2,-(SP)
2227          MOV    #BUF1,R1      ;FIRST BUFFER START
2228          MOV    #256.,R2      ;BUFFER COUNT
2229          1$: JSR    R5,RAND    ;GET RANDOM NUMBER
2230          MOV    LONUM,(R1)+    ;STORE IN BUFFER
2231          DEC    R2             ;CHECK FOR BUFFER END
2232          BNE    1$
2233          MOV    #BUF2,R1      ;SETUP TO CLEAR BUF2
2234          MOV    #255.,R2
2235          2$: CLR    (R1)+
2236          DEC    R2
2237          BNE    2$           ;CHECK FOR BUFFER END

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 43
 CVRLAC.P11 31-AUG-82 11:25 ROUTINE TO CHECK FOR CONTROLLER ERRORS

SEQ 0043

```

2249 016354 012721 123456      MOV    #123456,(R1)+ ;STORE IN LAST BUFFER WORD
2250 016360 012602      MOV    (SP)+,R2
2251 016362 012601      MOV    (SP)+,R1
2252 016364 000205      RTS     R5
2253
2254
2255      ;THIS ROUTINE IS A DOUBLE PRECISION PSEUDO RANDOM NUMBER GENERATOR
2256      ;WITH A RANGE OF 0 TO 2(+33)-1.
2257      ;CALL:
2258      ;      JSR    R5,RAND      ;CALL THE ROUTINE
2259      ;      RETURN              ;RETURN HERE THE RANDOM NUMBER
2260      ;      ;WILL BE IN HINUM,LONUM
2261 016366 010146      RAND: MOV    R1,-(SP) ;PUSH R1 ON STACK
2262 016370 010246      MOV    R2,-(SP) ;PUSH R2 ON STACK
2263 016372 010346      MOV    R3,-(SP) ;PUSH R3 ON STACK
2264 016374 013703 002344      MOV    LONUM,R3 ;SET R3 WITH LOW
2265 016400 013701 002342      MOV    HINUM,R1 ;SET R1 WITH HIGH
2266 016404 012702 177771      MOV    #-7,R2 ;SET SHIFT COUNTER
2267 016410 006303      1$: ASL    R3 ;SHIFT R3 LEFT AND
2268 016412 006101      ROL    R1 ;ROTATE CARRY INTO R1 AND
2269 016414 005202      INC    R2 ;CHECK FOR DONE
2270 016416 001374      BNE    1$ ;CONTINUE SHIFT LOOP
2271 016420 063703 002344      ADD    LONUM,R3 ;ADD NUMBER TO MAKE X 129
2272 016424 005501      ADC    R1 ;PROPOGATE CARRY
2273 016426 063701 002342      ADD    HINUM,R1 ;ADD NUMBER TO MAKE X 129
2274 016432 062703 001057      ADD    #1057,R3 ;ADD LOW CONSTANT
2275 016436 005501      ADC    R1 ;PROPOGATE CARRY
2276 016440 062701 047401      ADD    #47401,R1 ;ADD HIGH CONSTANT
2277 016444 010337 002344      MOV    R3,LONUM ;SAVE R3
2278 016450 010137 002342      MOV    R1,HINUM ;SAVE R1
2279 016454 012603      MOV    (SP)+,R3 ;POP STACK INTO R3
2280 016456 012602      MOV    (SP)+,R2 ;POP STACK INTO R2
2281 016460 012601      MOV    (SP)+,R1 ;POP STACK INTO R1
2282 016462 000205      RTS     R5 ;RETURN
2283
2284
2285      .SBTTL ROUTINE TO CALCULATE CRC
2286
2287      ;ROUTINE WILL CALCULATE A CRC-16 CRC ON A WORD OF
2288      ;1-16 BITS IN LENGTH, RESULT IS RETURNED IN "CALBCC"
2289      ;
2290      ;      CALL: JSR    R5,SIMBCC
2291      ;      .WORD      ;NUMBER OF BITS (1-16)
2292      ;      .WORD      ;DATA FOR CRC CALCULATION
2293      ;      .WORD      ;PREVIOUS OR STARTING CRC
2294      ;      ;          (SHOULD BE ZEROED FOR START)
2295      ;
2296      ;      ROUTINE USES R0,R1,R2
2297 016464 010046      SIMBCC: MOV    R0,-(SP) ;SAVE R0
2298 016466 010146      MOV    R1,-(SP) ;SAVE R1
2299 016470 010246      MOV    R2,-(SP) ;SAVE R2
2300 016472 012537 002264      MOV    (R5)+,TEMP2 ;GET NUMBER OF BITS
2301 016476 012537 002266      MOV    (R5)+,TEMP3 ;GET DATA FOR CRC CALCULATION
2302 016502 012537 002270      MOV    (R5)+,TEMP4 ;GET STARTING CRC
2303 016506 005037 002260      1$: CLR    BCCFBK ;
2304 016512 013700 002270      MOV    TEMP4,R0 ;GET PRESENT CRC

```

2305	016516	006037	002266		ROR	TEMP3		:ROTATE NEW DATA
2306	016522	005500			ADC	R0		:MERGE NEW WITH OLD
2307	016524	032700	000001		BIT	#1,R0		:BIT 0 SET
2308	016530	001402			BEQ	2\$		:IF NOT CONTINUE
2309	016532	005137	002260		COM	BCCFBK		
2310	016536	013700	002254	2\$:	MOV	XPOLY,R0		:GET CRC POLYNOMIAL (CRC-16)
2311	016542	005100			COM	R0		:COMPLIMENT POLYNOMIAL
2312	016544	040037	002260		BIC	R0,BCCFBK		
2313	016550	000241			CLC			:CLEAR CARRY
2314	016552	006037	002270		ROR	TEMP4		
2315	016556	013700	002260		MOV	BCCFBK,R0		
2316	016562	013701	002270		MOV	TEMP4,R1		
2317	016566	010102			MOV	R1,R2		
2318	016570	040100			BIC	R1,R0		
2319	016572	043702	002260		BIC	BCCFBK,R2		
2320	016576	050200			BIS	R2,R0		
2321	016600	043737	002254	002270	BIC	XPOLY,TEMP4		
2322	016606	050037	002270		BIS	R0,TEMP4		
2323	016612	005337	002264		DEC	TEMP2		
2324	016616	001333			BNE	1\$		
2325	016620	013737	002270	002262	MOV	TEMP4,CALBCC		
2326	016626	012602			MOV	(SP)+,R2		
2327	016630	012601			MOV	(SP)+,R1		
2328	016632	012600			MOV	(SP)+,R0		
2329	016634	000205			RTS	R5		:RETURN
2330								
2331								
2332								
2333								
2334								
2335								
2336								
2337	016636	005237	002246		TRPHAN: INC	TRPFLG		:INDICATE TRAP
2338	016642	000002			RTI			:RETURN
2339								
2340	016644				BGNSRV			
2341								
2342	016644	005237	002250		INTSRV: INC	INTFLG		:INDICATE INTERRUPT
2343								
2344	016650				ENDSRV			
2345	016650				L10022:			
2346	016650	000002			RTI			
2347								
2348								
2349	016652	005237	002250		:ROUTINE USED IN TIMING OPI			
2350	016656	000002			TIMSRV: INC	INTFLG		
2351					RTI			
2352								
2353								
2354	016660	010146			:ROUTINE TO WAIT FOR DRIVE READY			
2355	016662	012701	000310		WTD RDY: MOV	R1,-(SP)		:SAVE R1
2356	016666	032777	000001	163304	MOV	#200.,R1		:TIME OUT OF 200 MILLISECDS
2357	016674	001011			1\$: BIT	#DRDY,DRDCS		:DRIVE READY?
2358					BNE	2\$		:YES, EXIT
2359	016676	004537	016032		JSR	R5,WDELAY		:WAIT A WHILE
2360	016702	000001			1			:APPROX. A MILLISECOND

```

2361 016704 005301      DEC      R1      ;CHECK IF TIME UP
2362 016706 001367      BNE      1$      ;NO, GO CHECK DRIVE READY
2363
2364 016710      ERRDF      200.,DRTIM,ERR5 ;DRIVE READY DID NOT SET
2365 016710 104455      TRAP      C$ERDF
2366 016712 000310      .WORD      200
2367 016714 006547      .WORD      DRTIM
2368 016716 012074      .WORD      ERR5
2369
2370 016720 012601      2$:      MOV      (SP)+,R1      ;RESTORE
2371 016722 000205      RTS      R5      ;EXIT
2372
2373      ;ROUTINE TO WAIT FOR CONTROLLER READY
2374 016724 010146      WTCRDY: MOV      R1,-(SP)      ;SAVE R1
2375 016726 012701 001440      MOV      #800.,R1      ;WAIT 800 MILLISECONDS
2376 016732 032777 000200 163240 1$:      BIT      #CRDY,@RLCS      ;CONTROLLER READY
2377 016740 001014      BNE      2$      ;YES, EXIT
2378 016742 004537 016032      JSR      R5,WDELAY      ;WAIT A WHILE
2379 016746 000001      1      ;APPROX A MILLISECOND
2380 016750 005301      DEC      R1      ;CHECK IF TIME UP
2381 016752 001367      BNE      1$      ;NO GO BACK
2382
2383 016754 004537 015712      JSR      R5,AFTER      ;GET REGISTERS
2384
2385 016760      ERRDF      100.,CRTIM,ERR6 ;CONTROLLER TIMED OUT
2386 016760 104455      TRAP      C$ERDF
2387 016762 000144      .WORD      100
2388 016764 006522      .WORD      CRTIM
2389 016766 012106      .WORD      ERR6
2390
2391 016770 000402      BR      3$      ;EXIT
2392
2393 016772 004537 015712      2$:      JSR      R5,AFTER      ;GET REGISTERS
2394 016776 012601      3$:      MOV      (SP)+,R1
2395 017000 000205      RTS      R5      ;EXIT
2396
2397
2398
2399 017002      ENDMOD
2400
2401
2402
2403      .SBTTL  **TEST 1** - RLCS WRITE ADDRESSABILITY
2404
2405 017002      BGN1ST      ;****START OF TEST****
2406 017002      STARS
2407      ;*****
2408      ;TEST TO SEE IF WE CAN ADDRESS THE CONTROL
2409      ;AND STATUS REGISTER. IF WE TRAP WE WILL REPORT
2410      ;THE ERROR AND ABORT. AFTER THIS TEST WE ONLY KNOW
2411      ;THAT WE CAN ADDRESS THE REGISTER.
2412 017002      STARS
2413      ;*****
2414
2415
2416 017002 005037 002246      1$:      CLR      TRPFLG      ;CLEAR TRAP OCCURANCE
  
```

```
2417 017006          2$: SETVEC ERRVEC,#TRPHAN,#340 ;SET TO CATCH TRAP
2418 017006 012746 000340      MOV #340,-(SP)
2419 017012 012746 016636      MOV #TRPHAN,-(SP)
2420 017016 013746 002256      MOV ERRVEC,-(SP)
2421 017022 012746 000003      MOV #3,-(SP)
2422 017026 104437          TRAP C$SVEC
2423 017030 062706 000010      ADD #10,SP
2424
2425 017034 012777 177777 163136  MOV #177777,@RLCS ;ADDRESS RLCS
2426 017042          CLRVEC ERRVEC ;RELEASE TRAP VECTOR
2427 017042 013700 002256      MOV ERRVEC,R0
2428 017046 104436          TRAP C$CVEC
2429 017050 005737 002246      TST TRPFLG ;TRAP OCCURRED???
2430 017054 001407          BEQ 3$ ;NO, OKAY PROCEED
2431 017056 013737 002200 002306  MOV RLCS,GDDAT ;SET UP ERROR DATA
2432
2433 017064          ERRSF 0,EM1,ERR1 ;BUS TIMEOUT IN ADDRESSING RLCS
2434 017064 104454          TRAP C$ERSF
2435 017066 000000          .WORD 0
2436 017070 006575          .WORD EM1
2437 017072 011650          .WORD ERR1
2438 017074          3$: CKLOOP ;CHECK IF /FL:LOE IS SET
2439 017074 104406          TRAP C$CLP1
2440 017076          ENDTST ;****END OF TEST****
2441 017076          L10023:
2442 017076 104401          TRAP C$ETST
2443
2444
2445
2446
2447 017100          .SBTTL **TEST 2** - RLBA WRITE ADDRESSABILITY
2448          BGNTST ;****START OF TEST****
2449
2450 017100          STARS
2451          ;*****
2452          ;TEST TO SEE IF WE CAN ADDRESS THE BUS ADDRESS
2453          ;REGISTER. IF WE TRAP WE WILL REPORT THE ERROR
2454          ;AND ABORT. AFTER THIS TEST WE ONLY KNOW THAT
2455          ;WE CAN ADDRESS THE REGISTER.
2456 017100          STARS
2457          ;*****
2458
2459 017100 005037 002246          1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2460 017104          2$: SETVEC ERRVEC,#TRPHAN,#340 ;SET TO CATCH TRAP
2461 017104 012746 000340      MOV #340,-(SP)
2462 017110 012746 016636      MOV #TRPHAN,-(SP)
2463 017114 013746 002256      MOV ERRVEC,-(SP)
2464 017120 012746 000003      MOV #3,-(SP)
2465 017124 104437          TRAP C$SVEC
2466 017126 062706 000010      ADD #10,SP
2467
2468 017132 012777 177777 163042  MOV #177777,@RLBA ;ADDRESS RLBA
2469 017140          CLRVEC ERRVEC ;RELEASE TRAP VECTOR
2470 017140 013700 002256      MOV ERRVEC,R0
2471 017144 104436          TRAP C$CVEC
2472 017146 005737 002246      TST TRPFLG ;TRAP OCCURRED???
```

```

2473 017152 001407      BEQ      3$      :NO, CONTINUE
2474 017154 013737 002202 002306      MOV      RLBA,GDDAT      :SETUP ERROR DATA
2475
2476 017162      ERRSF      1,EM2,ERR1      :BUS TIMEOUT IN ADDRESSING RLBA
2477 017162 104454      TRAP      C$ERSF
2478 017164 000001      .WORD      1
2479 017166 006622      .WORD      EM2
2480 017170 011650      .WORD      ERR1
2481 017172      3$:      CKLOOP      :CHECK IF /FL:LOE IS SET
2482 017172 104406      TRAP      C$CLP1
2483 017174      ENDTST      :****END OF TEST****
2484 017174      L10024:
2485 017174 104401      TRAP      C$ETST
2486
2487
2488
2489

```

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

```

2490 017176      BGNST      :****START OF TEST****
2491 017176      STARS
2492      :*****
2493      :TEST TO SEE IF WE CAN ADDRESS THE DISK ADDRESS
2494      :REGISTER IF WE TRAP WE WILL REPORT THE ERROR
2495      :AND ABORT. AFTER THIS TEST WE ONLY KNOW THAT
2496      :WE CAN ADDRESS THE REGISTER.
2497 017176      STARS
2498      :*****
2499
2500

```

```

2501 017176 005037 002246      1$:      CLR      TRPFLG      :CLEAR TRAP OCCURANCE
2502 017202      2$:      SETVEC      ERRVEC,#TRPHAN,#340 ;SET TO CATCH TRAP
2503 017202 012746 000340      MOV      #340,-(SP)
2504 017206 012746 016636      MOV      #TRPHAN,-(SP)
2505 017212 013746 002256      MOV      ERRVEC,-(SP)
2506 017216 012746 000003      MOV      #3,-(SP)
2507 017222 104437      TRAP      C$SVEC
2508 017224 062706 000010      ADD      #10,SP
2509
2510 017230 012777 177777 162746      MOV      #177777,@RLDA      :ADDRESS RLDA
2511 017236      CLRVEC      ERRVEC      :RELEASE TRAP VECTOR
2512 017236 013700 002256      MOV      ERRVEC,R0
2513 017242 104436      TRAP      C$CVEC
2514 017244 005737 002246      TST      TRPFLG      :TRAP OCCURRED???
2515 017250 001407      BEQ      3$      :NO, CONTINUE
2516
2517 017252 013737 002204 002306      MOV      RLDA,GDDAT      :SETUP ERROR INFO
2518 017260      ERRSF      2,EM3,ERR1      :BUS TIMEOUT IN ADDRESSING RLDA
2519 017260 104454      TRAP      C$ERSF
2520 017262 000002      .WORD      2
2521 017264 006647      .WORD      EM3
2522 017266 011650      .WORD      ERR1
2523 017270      3$:      CKLOOP      :CHECK IF /FL:LOE IS SET
2524 017270 104406      TRAP      C$CLP1
2525 017272      ENDTST      :****END OF TEST****
2526 017272      L10025:
2527 017272 104401      TRAP      C$ETST
2528

```

```
2529
2530
2531
2532 017274
2533 017274
2534
2535
2536
2537
2538
2539 017274
2540
2541
2542
2543 017274 005037 002246
2544 017300
2545 017300 012746 000340
2546 017304 012746 016636
2547 017310 013746 002256
2548 017314 012746 000003
2549 017320 104437
2550 017322 062706 000010
2551
2552 017326 012777 177777 162652
2553 017334
2554 017334 013700 002256
2555 017340 104436
2556 017342 005737 002246
2557 017346 061407
2558 017350 013737 002206 002306
2559
2560 017356
2561 017356 104454
2562 017360 000003
2563 017362 006674
2564 017364 011650
2565 017366
2566 017366 104406
2567 017370
2568 017370
2569 017370 104401
2570
2571
2572
2573 017372
2574 017372
2575
2576
2577
2578
2579
2580 017372
2581
2582
2583
2584 017372 005037 002246

.SBTTL **TEST 4** - RLMP WRITE ADDRESSABILITY
BGNIST
STARS
*****
;TEST TO SEE IF WE CAN ADDRESS THE MULTIPURPOSE
;REGISTER. IF WE TRAP WE WILL REPORT THE ERROR AND
;ABORT. AFTER THIS TEST WE ONLY KNOW THAT WE CAN
;ADDRESS THE REGISTER.
STARS
*****
1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2$: SETVEC ERRVEC,TRPHAN,#340 ;SET UP TO CATCH TRAP
MOV #340,-(SP)
MOV TRPHAN,-(SP)
MOV ERRVEC,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP
MOV #177777,@RLMP ;ADDRESS RLMP
CLRVEC ERRVEC ;RELEASE TRAP VECTOR
MOV ERRVEC,R0
TRAP C$CVEC
TST TRPFLG ;TRAP OCCURRED???
BEQ 3$ ;NO, CONTINUE
MOV RLMP,GDDAT ;SET UP ERROR INFO
ERRSF 3,EM4,ERR1 ;BUS TIMEOUT IN ADDRESSING RLMP
TRAP C$ERSF
.WORD 3
.WORD EM4
.WORD ERR1
3$: CKLOOP ;CHECK IF /FL:LOE IS SET
TRAP C$CLP1
ENDIST
L10026: TRAP C$ETST

.SBTTL **TEST 5** - RLCS READ ADDRESSABILITY
BGNIST
STARS
*****
;TEST TO SEE IF WE CAN ADDRESS THE CONTROL
;AND STATUS REGISTER. IF WE TRAP WE WILL REPORT
;THE ERROR AND ABORT. AFTER THIS TEST WE ONLY KNOW
;THAT WE CAN ADDRESS THE REGISTER.
STARS
*****
1$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
```

2585	017376			2\$:	SETVEC	ERRVEC,#TRPHAN,#340	;SET TO CATCH TRAP
2586	017376	012746	000340		MOV	#340,-(SP)	
2587	017402	012746	016636		MOV	#TRPHAN,-(SP)	
2588	017406	013746	002256		MOV	ERRVEC,-(SP)	
2589	017412	012746	000003		MOV	#3,-(SP)	
2590	017416	104437			TRAP	C\$SVEC	
2591	017420	062706	000010		ADD	#10,SP	
2592							
2593	017424	005777	162550		TST	@RLCS	;ADDRESS RLCS
2594	017430				CLRVEC	ERRVEC	;RELEASE TRAP VECTOR
2595	017430	013700	002256		MOV	ERRVEC,R0	
2596	017434	104436			TRAP	C\$CVEC	
2597	017436	005737	002246		TST	TRPFLG	;TRAP OCCURRED???
2598	017442	001407			BEQ	3\$	;NO, OKAY PROCEED
2599	017444	013737	002200 002306		MOV	RLCS,GDDAT	;SET UP ERROR DATA
2600							
2601	017452				ERRSF	100,EM1,ERR1	;BUS TIMEOUT IN ADDRESSING RLCS
2602	017452	104454			TRAP	C\$ERSF	
2603	017454	000144			.WORD	100	
2604	017456	006575			.WORD	EM1	
2605	017460	011650			.WORD	ERR1	
2606	017462			3\$:	CKLOOP		;CHECK IF /FL:LOE IS SET
2607	017462	104406			TRAP	C\$CLP1	
2608	017464			ENDTST			;****END OF TEST****
2609	017464			L10027:			
2610	017464	104401			TRAP	C\$ETST	
2611							
2612							
2613							
2614							
2615	017466						
2616							
2617							
2618	017466						
2619							
2620							
2621							
2622							
2623							
2624	017466						
2625							
2626							
2627	017466	005037	002246	1\$:	CLR	TRPFLG	;CLEAR TRAP OCCURANCE
2628	017472			2\$:	SETVEC	ERRVEC,#TRPHAN,#340	;SET TO CATCH TRAP
2629	017472	012746	000340		MOV	#340,-(SP)	
2630	017476	012746	016636		MOV	#TRPHAN,-(SP)	
2631	017502	013746	002256		MOV	ERRVEC,-(SP)	
2632	017506	012746	000003		MOV	#3,-(SP)	
2633	017512	104437			TRAP	C\$SVEC	
2634	017514	062706	000010		ADD	#10,SP	
2635							
2636	017520	005777	162456		TST	@RLBA	;ADDRESS RLBA
2637	017524				CLRVEC	ERRVEC	;RELEASE TRAP VECTOR
2638	017524	013700	002256		MOV	ERRVEC,R0	
2639	017530	104436			TRAP	C\$CVEC	
2640	017532	005737	002246		TST	TRPFLG	;TRAP OCCURRED???

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

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

STARS

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

1\$:	CLR	TRPFLG	;CLEAR TRAP OCCURANCE
2\$:	SETVEC	ERRVEC,#TRPHAN,#340	;SET TO CATCH TRAP
	MOV	#340,-(SP)	
	MOV	#TRPHAN,-(SP)	
	MOV	ERRVEC,-(SP)	
	MOV	#3,-(SP)	
	TRAP	C\$SVEC	
	ADD	#10,SP	
	TST	@RLBA	;ADDRESS RLBA
	CLRVEC	ERRVEC	;RELEASE TRAP VECTOR
	MOV	ERRVEC,R0	
	TRAP	C\$CVEC	
	TST	TRPFLG	;TRAP OCCURRED???

```

2641 017536 001407      BEQ      3$      ;NO, CONTINUE
2642 017540 013737 002202 002306      MOV      RLBA,GDDAT      ;SETUP ERROR DATA
2643
2644 017546      ERRSF      101.,EM2,ERR1      ;BUS TIMEOUT IN ADDRESSING RLBA
2645 017546 104454      TRAP      C$ERSF
2646 017550 000145      .WORD      101
2647 017552 006622      .WORD      EM2
2648 017554 011650      .WORD      ERR1
2649 017556      3$:      CKLOOP      ;CHECK IF /FL:LOE IS SET
2650 017556 104406      TRAP      C$CLP1
2651 017560      ENDTST      ;****END OF TEST****
2652 017560      L10030:
2653 017560 104401      TRAP      C$ETST
2654
2655
2656
2657

```

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

```

2658 017562      BGNSTST      ;****START OF TEST****
2659 017562      STARS
2660      ;*****
2661      ;TEST TO SEE IF WE CAN ADDRESS THE DISK ADDRESS
2662      ;REGISTER IF WE TRAP WE WILL REPORT THE ERROR
2663      ;AND ABORT. AFTER THIS TEST WE ONLY KNOW THAT
2664      ;WE CAN ADDRESS THE REGISTER.
2665 017562      STARS
2666      ;*****
2667
2668
2669 017562 005037 002246      1$:      CLR      TRPFLG      ;CLEAR TRAP OCCURANCE
2670 017566      2$:      SETVEC      ERRVEC,#TRPHAN,#340 ;SET TO CATCH TRAP
2671 017566      MOV      #340,-(SP)
2672 017572 012746 000340      MOV      #TRPHAN,-(SP)
2673 017576 013746 002256      MOV      ERRVEC,-(SP)
2674 017602 012746 000003      MOV      #3,-(SP)
2675 017606 104437      TRAP      C$SVEC
2676 017610 062706 000010      ADD      #10,SP
2677
2678 017614 005777 162364      TST      @RLDA      ;ADDRESS RLDA
2679 017620      CLRVEC      ERRVEC      ;RELEASE TRAP VECTOR
2680 017620 013700 002256      MOV      ERRVEC,R0
2681 017624 104436      TRAP      C$CVEC
2682 017626 005737 002246      TST      TRPFLG      ;TRAP OCCURRED???
2683 017632 001407      BEQ      3$      ;NO, CONTINUE
2684
2685 017634 013737 002204 002306      MOV      RLDA,GDDAT      ;SETUP ERROR INFO
2686 017642      ERRSF      102.,EM3,ERR1      ;BUS TIMEOUT IN ADDRESSING RLDA
2687 017642 104454      TRAP      C$ERSF
2688 017644 000146      .WORD      102
2689 017646 006647      .WORD      EM3
2690 017650 011650      .WORD      ERR1
2691 017652      3$:      CKLOOP      ;CHECK IF /FL:LOE IS SET
2692 017652 104406      TRAP      C$CLP1
2693 017654      ENDTST      ;****END OF TEST****
2694 017654      L10031:
2695 017654 104401      TRAP      C$ETST
2696

```

2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752

017656
017656

017656

017656 005037 002246

017662

017662 012746 000340

017666 012746 016636

017672 013746 002256

017676 012746 000003

017702 104437

017704 062706 000010

017710 005777 162272

017714

017714 013700 002256

017720 104436

017722 005737 002246

017726 001407

017730 013737 002206 002306

017736

017736 104454

017740 000147

017742 006674

017744 011650

017746

017746 104406

017750

017750

017750 104401

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

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

STARS

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

STARS

1\$: CLR TRPFLG ;CLEAR TRAP OCCURANCE
2\$: SETVEC ERRVEC,#TRPHAN,#340 ;SET UP TO CATCH TRAP

MOV #340,-(SP)
MOV #TRPHAN,-(SP)
MOV ERRVEC,-(SP)
MOV #3,-(SP)
TRAP C\$SVEC
ADD #10,SP

TST @RLMP ;ADDRESS RLMP
CLRVEC ERRVEC ;RELEASE TRAP VECTOR
MOV ERRVEC,R0
TRAP C\$CVEC
TST TRPFLG ;TRAP OCCURRED??
BEQ 3\$;NO, CONTINUE
MOV RLMP,GDDAT ;SET UP ERROR INFO

ERRSF 103,EM4,ERR1 ;BUS TIMEOUT IN ADDRESSING RLMP
TRAP C\$ERSF
.WORD 103
.WORD EM4
.WORD ERR1

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

TRAP C\$CLP1

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

L10032:

TRAP C\$ETST

.SBTTL **TEST 9** - BUS RESET OF RLCS

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

STARS

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

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 52
 (VRLAC.P11 31-AUG-82 11:25 **TEST 9** - BUS RESET OF RLCS

SEQ 0052

```

2753 017752 STARS
2754 ;:*****
2755
2756
2757 017752 SETPRI #PRI07 ;PRIORITY TO SEVEN
2758 017752 MOV #PRI07,R0
2759 017756 TRAP C$SPRI
2760 017760 MOV #377,@RLCS ;LOAD ALL RLCS LOADABLE BITS
2761 017766 MOV #CRDY,GDDAT ;SETUP EXPECTED
2762 017774 032777 040000 162176 BIT #DERR,@RLCS ;DRIVE ERR SET?
2763 020002 001403 BEQ 1$ ;IF NOT DON'T EXPECT IT
2764 020004 052737 140000 002306 BIS #DERR!ERR,GDDAT ;IT'S SET, INIT BETTER NOT CLR
2765 020012 012700 0C0100 1$: MOV #100,R0 ;SET UP A WAIT LOOP
2766 020016 BRESET ;BUS RESET
2767 020016 104433 TRAP C$RESET
2768 020020 005300 2$: DEC R0 ;WAIT IN CASE OF DRIVE ERROR
2769 020022 001376 BNE 2$
2770 020024 017737 162150 002310 MOV @RLCS,BDDAT ;READ RLCS
2771 020032 042737 000001 002310 BIC #DRDY,BDDAT ;CLEAR OUT DRDY - DON'T CARE
2772 020040 023737 002310 002306 CMP BDDAT,GDDAT ;DID INIT WORK
2773 020046 001404 BEQ 3$ ;YES, BRANCH
2774
2775 020050 ERRDF 113,EM67,ERR2 ;WRONG DATA IN RLCS
2776 020050 104455 TRAP C$ERDF
2777 020052 000161 .WORD 113
2778 020054 011005 .WORD EM67
2779 020056 011662 .WORD ERR2
2780
2781 020060 3$:
2782 020060 ENDTST ;****END OF TEST****
2783 020060 104401 L10033: TRAP C$ETST
2784
2785
2786 .SBTTL **TEST 10** - BUS RESET OF RLBA
2787
2788 020062 BGNST ;****START OF TEST****
2789
2790 020062 STARS
2791 ;:*****
2792 ;TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
2793 ;BUS ADDRESS REGISTER. THE BUS ADDRESS IS LOADED WITH 177776
2794 ;AND IS EXPECTED TO BE ZERO AFTER THE RESET
2795 020062 STARS
2796 ;:*****
2797
2798
2799 020062 012777 177776 162112 MOV #-2,@RLBA ;SET BA TO ALL 1'S
2800 020070 005737 002332 TST 1,CNTRL ;RL11?
2801 020074 001403 BEQ 2$ ;NO
2802 020076 052777 000001 162076 BIS #1,@RLBA
2803 020104 005037 002306 2$: CLR GDDAT ;CLEAR EXPECTED DATA
2804 020110 BRESET ;ISSUE BUS INIT
2805 020110 104433 TRAP C$RESET
2806 020112 017737 162064 002310 MOV @RLBA,BDDAT ;READ RLBA
2807 020120 001404 BEQ 1$ ;IF CLEAR BRANCH
2808

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 53
 FURLAC.P11 31-AUG-82 11:25 **TEST 10** - BUS RESET OF RLBA

SEQ 0053

```

2809 020122 ERRDF 114.,EM70,ERR2 ;WRONG DATA IN RLBA
2810 020122 104455 TRAP C$ERDF
2811 020124 000162 .WORD 114
2812 020126 011042 .WORD EM70
2813 020130 011662 .WORD ERR2
2814 020132 1$:
2815
2816 020132 ENDTST ;****END OF TEST****
2817 020132 L10034:
2818 020132 104401 TRAP C$ETST
2819
2820
2821 .SBTTL **TEST 11** - BUS RESET OF RLDA
2822
2823 020134 BGNTST ;****START OF TEST****
2824
2825 020134 STARS
2826 ;*****
2827 ;TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
2828 ;DISK ADDRESS REGISTER. THE DISK ADDRESS IS LOADED WITH 177777
2829 ;AND IS EXPECTED TO BE ZERO AFTER THE RESET.
2830 020134 STARS
2831 ;*****
2832
2833
2834 020134 012777 177777 162042 MOV #-1,@RLDA ;SET DA TO ALL 1'S
2835 020142 005037 002306 CLR GDDAT ;CLEAR EXPECTED
2836 020146 @RESET ;ISSUE BUS INIT
2837 020146 104433 TRAP C$RESET
2838 020150 017737 162030 002310 MOV @RLDA,BDDAT ;READ RLDA
2839 020156 001404 BEQ 1$ ;IF CLEAR BRANCH
2840
2841 020160 ERRDF 115.,EM71,ERR2 ;WRONG DATA IN RLDA
2842 020160 104455 TRAP C$ERDF
2843 020162 000163 .WORD 115
2844 020164 011077 .WORD EM71
2845 020166 011662 .WORD ERR2
2846 020170 1$:
2847
2848 020170 ENDTST ;****END OF TEST****
2849 020170 L10035:
2850 020170 104401 TRAP C$ETST
2851
2852
2853 .SBTTL **TEST 12** - READ WRITE OF RLCS
2854
2855 020172 BGNTST ;****START OF TEST****
2856
2857
2858
2859 020172 STARS
2860 ;*****
2861 ;TEST THAT WE CAN WRITE/READ BITS 8,9 AND BITS 6-1
2862 ;OF THE CONTROL AND STATUS REGISTER. BITS 15-10 AND 0
2863 ;ARE DON'T CARE BITS AT THIS TIME AND BIT 7
2864 ; (CONTROLLER READY) IS ALWAYS WRITTEN TO A ONE.

```

MAIN. MACY11 30(1046) 31-AUG-82 1:50 PAGE 54
 CVRLAC.P11 31-AUG-82 11:25 **TEST 12** - READ WRITE OF RLCS

SEQ 0054

```

2865 020172          STARS
2866                ;*****
2867
2868
2869 020172 012703 003160      MOV      #CSPAT,R3      ;SET UP TABLE POINTER OF PATTERNS
2870
2871 020176          BGNSEG          ;****START OF SEGMENT****
2872 020176 104404          TRAP      C$BSEG
2873
2874 020200          C$TEST:
2875 020200 011337 002306      MOV      (R3),GDDAT      ;GET PATTERN INTO GDDAT
2876 020204 052737 000200 002306      BIS      #200,GDDAT      ;INSURE GO IS SET
2877 020212 013777 002306 161760      MOV      GDDAT,@RLCS      ;LOAD RLCS (CONTROL AND STATUS)
2878 020220 032777 040000 161752      BIT      #DERR,@RLCS      ;IF DRIVE ERROR PRESENT
2879 020226 001403          BEQ      99$      ;THEN EXPECT DRIVE AND
2880 020230 052737 140000 002306      BIS      #ERR!DERR,GDDAT      ;COMPOSITE ERROR
2881 020236 017737 161736 002310 99$:      MOV      @RLCS,BDDAT      ;READ RLCS BACK
2882 020244 042737 000001 002310      BIC      #DRDY,BDDAT      ;IGNORE DRIVE READY
2883 020252 023737 002306 002310      CMP      GDDAT,BDDAT      ;DID WE READ WHAT WE LOADED
2884 020260 001404          BEQ      1$      ;YES, THEN BRANCH
2885
2886 020262          ERRDF 4,,EM5,ERR2      ;WRONG DATA IN RLCS
2887 020262 104455          TRAP      C$ERDF
2888 020264 000004          .WORD      4
2889 020266 006721          .WORD      EM5
2890 020270 011662          .WORD      ERR2
2891 020272          1$:      ESCAPE SEG      ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
2892 020272 104410          TRAP      C$ESCAPE
2893 020274 000012          .WORD      10000$-.
2894
2895
2896 020276 005723          TST      (R3)+      ;BUMP FOR NEXT PATTERN
2897 020300 020327 003256      CMP      R3,#CSEND      ;CHECK FOR END
2898 020304 001335          BNE      C$TEST      ;NOT END, LOAD NEXT PATTERN
2899
2900          ENDSEG          ;****END OF SEGMENT****
2901 020306 10000$:      TRAP      C$ESEG
2902 020306 104405
2903 020310          ENDTST          ;****END OF TEST****
2904 020310 L10036:
2905 020310 104401          TRAP      C$ETST
2906
2907
2908          .SBTTL  **TEST 13** - READ WRITE OF RLBA
2909
2910 020312          BGNTEST          ;****START OF TEST****
2911
2912 020312          STARS
2913                ;*****
2914                ;TEST THAT WE CAN WRITE/READ BITS IS THRU 1 OF THE
2915                ;BUS ADDRESS REGISTER. FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
2916                ;GROWING 0 AND SHIFTING 0. BIT 0 IS ALSO LOADED BUT
2917                ;SHOULD ALWAYS COME BACK AS 0
2918 020312          STARS
2919                ;*****
2920

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 55
 (VRLAC.P11 31-AUG-82 11:25 **TEST 13** - READ WRITE OF RLBA

SEQ 0055

```

2921
2922 020312 012703 002564      BGNSEG MOV    #BEGPAT,R3      ;GET START OF PATTERN LIST
2923 020316      TRAP    C$BSEG      ;****START OF SEGMENT****
2924 020316 104404
2925 020320      BATEST:
2926 020320 011337 002306      MOV    (R3),GDDAT      ;GET PATTERN TO SEND
2927 020324 005737 002332      TST    T,CN1LR      ;RL11??
2928 020330 001403      BEQ    2$      ;NO
2929 020332 042737 000001 002306      BIC    #BIT0,GDDAT      ;KEEP RLBA EVEN (UNIBUS)
2930 020340 013777 002306 161634 2$: MOV    GDDAT,@RLBA      ;LOAD PATTERN TO BUS ADDRESS
2931 020346 017737 161630 002310      MOV    @RLBA,BDDAT      ;READ IT BACK
2932 020354 023737 002306 002310      CMP    GDDAT,BDDAT      ;IS IT CORRECT?
2933 020362 001404      BEQ    1$      ;IF SO, BRANCH
2934
2935 020364      ERRDF    5,,EM6,ERR2      ;DATA WRONG IN RLBA
2936 020364 104455      TRAP    C$ERDF
2937 020366 000005      .WORD    5
2938 020370 006772      .WORD    EM6
2939 020372 011662      .WORD    ERR2
2940 020374      1$: ESCAPE SEG
2941 020374 104410      TRAP    C$ESCAPE
2942 020376 000012      .WORD    10000$-.
2943
2944 020400 005723      TST    (R3)+      ;BUMP FOR NEXT PATTERN
2945 020402 020327 002772      CMP    R3,#ENDPAT      ;CHECK FOR END
2946 020406 001344      BNE    BATEST      ;NOT END, BRANCH FOR NEXT
2947
2948 020410      ENDSEG      ;****END OF SEGMENT****
2949 020410 10000$:
2950 020410 104405      TRAP    C$ESEG
2951 020412      ENDTST      ;****END OF TEST****
2952 020412 L10037:
2953 020412 104401      TRAP    C$ETST
2954
2955
2956      .SBTTL  **TEST 14** - READ WRITE OF RLDA
2957
2958 020414      BGNTEST      ;****START OF TEST****
2959
2960 020414      STARS
2961      ;*****
2962      ;TEST THAT WE CAN WRITE/READ THE DISK ADDRESS REGISTER
2963      ;ALL BIT POSITIONS ARE WRITTEN USING FOUR PATTERNS:
2964      ;GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
2965 020414      STARS
2966      ;*****
2967
2968
2969 020414 012703 002564      BGNSEG MOV    #BEGPAT,R3      ;SET UP POINTER TO PATTERN LIST
2970 020420      TRAP    C$BSEG      ;****START OF SEGMENT****
2971 020420 104404
2972 020422      DATEST:
2973 020422 011337 002306      MOV    (R3),GDDAT      ;GET PATTERN
2974 020426 013777 002306 161550      MOV    GDDAT,@RLDA      ;LOAD PATTERN IN DA
2975
2976 020434 017737 161544 002310      MOV    @RLDA,BDDAT      ;READ PATTERN BACK

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 56
 VRLAC.P11 31-AUG-82 11:25 **TEST 14** - READ WRITE OF RLDA

SEQ 0056

```

2977 020442 023737 002306 002310      CMP      GDDAT,BDDAT      ;IS IT CORRECT?
2978 020450 001404                      BEQ      1$              ;BRANCH IF CORRECT
2979
2980 020452                      ERRDF      6,EM7,ERR2      ;WRONG DATA IN RLDA
2981 020452 104455                      TRAP      C$ERRDF
2982 020454 000006                      .WORD      6
2983 020456 007020                      .WORD      EM7
2984 020460 011662                      .WORD      ERR2
2985 020462                      1$:      ESCAPE      SEG      ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
2986 020462 104410                      TRAP      C$ESCAPE
2987 020464 000012                      .WORD      10000$-
2988
2989
2990 020466 005723 002772      TST      (R3)+      ;BUMP POINTER
2991 020470 020327                      CMP      R3,#ENDPAT  ;AT END OF PATTERNS?
2992 020474 001352                      BNE      DATEST      ;NO, BRANCH BACK
2993
2994 020476                      ENDSEG                      ;****END OF SEGMENT****
2995 020476 10000$:
2996 020476 104405      TRAP      C$ESEG
2997 020500                      ENDTST                      ;****END OF TEST****
2998 C20500  L10040:
2999 020500 104401      TRAP      C$ETST
3000
3001
3002      .SBTTL  **TEST 15** - BIS OF RLCS
3003
3004 020502      BGNSTST                      ;****START OF TEST****
3005 020502      STARS
3006      ;*****
3007      ;TEST THAT WE CAN USE THE "BIS" INSTRUCTION ON THE CONTROL
3008      ;AND STATUS REGISTER. BITS 8,9 AND 6-1 ARE TESTED TO
3009      ;SET INDIVIDUALLY AS WELL AS COLLECTIVELY WITHOUT DESTROYING
3010      ;ANY PREVIOUS DATA PATTERN
3011 020502      STARS
3012      ;*****
3013
3014
3015 020502 012703 003160      BGNSEG      MOV      #CSPAT,R3      ;GET BEGINNING OF LIST
3016 020506                      TRAP      C$BSEG      ;****START OF SEGMENT****
3017 020506 104404      1$:
3018 020510                      MOV      #CRDY,@RLCS      ;INSURE GO IS THERE
3019 020510 012777 000200 161462      MOV      (R3),GDDAT      ;SET UP EXPECTED RLCS
3020 020516 011337 002306      BIS      #CRDY,GDDAT      ;IN GDDAT
3021 020522 052737 000200 002306      BIS      (R3),@RLCS      ;BIT SET PATTERN IN RLCS
3022 020530 051377 161444      BIT      #DERR,@RLCS      ;IF ERROR BIT SET THEN
3023 020534 032777 040000 161436      BEQ      99$          ;EXPECT IT ON THE READ
3024 020542 001403                      BIS      #ERR!DERR,GDDAT ;BACK
3025 020544 052737 140000 002306      99$:      MOV      @RLCS,BDDAT      ;READ RLCS TO CHECK "BIS"
3026 020552 017737 161422 002310      BIC      #DRDY,BDDAT      ;CLEAR OUT DRIVE READY
3027 020560 042737 000001 002310      CMP      BDDAT,GDDAT      ;DID BIS WORK?
3028 020566 023737 002310 002306      BEQ      2$          ;BRANCH IF OKAY
3029 020574 001404
3030
3031 020576                      ERRDF      7,EM61,ERR2      ;WRONG DATA IN RLCS
3032 020576 104455                      TRAP      C$ERRDF

```

MAIN. MAR 11 30(1046) 31-AUG-82 11:50 PAGE 57
 (VRLAC.P11 31-AUG-82 11:25 **TEST 15** - BIS OF RLCS

SEQ 0057

```

3033 020600 000007      .WORD 7
3034 020602 010331      .WORD EM61
3035 020604 011662      .WORD ERR2
3036 020606 2$:        ESCAPE SEG      ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
3037 020606 104410      TRAP C$ESCAPE
3038 020610 000012      .WORD 10000$-
3039                                ;BIT OR CLEARED OTHER BIT
3040
3041 020612 005723      TST (R3)+      ;GET NEXT PATTERN
3042 020614 022703 003256 CMP #CSEND,R3 ;AT END OF LIST
3043 020620 001333      BNE 1$          ;NO GO BACK FOR TEST OF
3044                                ;NEXT PATTERN
3045                                ;****END OF SEGMENT****
3046 020622      ENDSEG
3047 020622 104405      10000$: TRAP C$ESEG
3048 020624      ENDTST
3049 020624 104401      L10041: TRAP C$ETST
3050
3051                                ;****END OF TEST****
3052
3053      .SBTTL **TEST 16** - BIC OF RLCS
3054
3055 020626      BGNSTST      ;****START OF TEST****
3056
3057 020626      STARS
3058      ;*****
3059      ;TEST THAT THE 'BIC' INSTRUCTION WILL WORK ON THE
3060      ;CONTROL AND STATUS REGISTER. BITS 8-9 AND 6-1 ARE
3061      ;TESTED.
3062 020626      STARS
3063      ;*****
3064
3065
3066 020626 012703 003160      BGNSEG MOV #CSPAT,R3      ;GET BEGINNING OF PATTERNS
3067 020632      TRAP C$BSEG      ;****START OF SEGMENT****
3068 020632 104404
3069 020634 1$:
3070 020634 012777 001776 161336 MOV #1776,@RLCS      ;SET ALL SETTABLE BITS
3071 020642 012737 001776 002306 MOV #1776,GDDAT      ;SET UP EXPECT DATA IN
3072 020650 041337 002306      BIC (R3),GDDAT      ;GDDAT
3073 020654 041377 161320      BIC (R3),@RLCS      ;CLEAR BITS IN RLCS VIA 'BIC'
3074 020660 032777 040000 161312 BIT #DERR,@RLCS      ;IF DRIVE ERROR BIT SET
3075 020666 001403      BEQ 99$      ;EXPECT IT SET WHEN WE
3076 020670 052737 140000 002306 BIS #ERR!DERR,GDDAT ;READ IT BACK
3077 020676 017737 161276 002310 99$: MOV @RLCS,BDDAT ;MOVE RLCS TO BDDAT FOR COMPARE
3078 020704 042737 000001 002310 BIC #DRDY,BDDAT ;CLEAR DRIVE READY
3079 020712 023737 002310 002306 CMP BDDAT,GDDAT ;DID 'BIC' WORK PROPERLY
3080 020720 001404      BEQ 2$      ;BRANCH IF OKAY
3081
3082 020722      ERRDF 8,EM62,ERR2 ;WRONG DATA IN RLCS
3083 020722 104455      TRAP C$ERDF
3084 020724 000010      .WORD 8
3085 020726 010412      .WORD EM62
3086 020730 011662      .WORD ERR2
3087 020732 2$:        ESCAPE SEG      ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
3088 020732 104410      TRAP C$ESCAPE

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 58
 CVRLAC.P11 31-AUG-82 11:25 **TEST 16** - BIC OF RLCS

SEQ 0058

```

3089 020734 000012 .WORD 10000$-.
3090
3091 020736 005723 TST (R3)+ ;GET NEXT PATTERN
3092 020740 020327 003256 CMP R3,#CSEND ;AT END OF LIST
3093 020744 001333 BNE 1$ ;NO, GO BACK WITH NEXT PATTERN
3094 020746 ENDSEG ;****END OF SEGMENT****
3095 020746 10000$:
3096 020746 104405 TRAP C$ESEG
3097 020750 ENDTST ;****END OF TEST****
3098 020750 L10042:
3099 020750 104401 TRAP C$ETST
3100
3101 .SBTTL **TEST 17** - BIS OF RLBA
3102
3103 BGNTEST ;****START OF TEST****
3104 020752
3105
3106 020752
3107 STARS
3108 ;*****
3109 ;TEST THAT THE "BIS" INSTRUCTION WILL WORK ON THE BUS
3110 ;ADDRESS REGISTER. BITS 15-0 ARE LOADED, ONLY BITS 15-1
3111 ;ARE EXPECTED BACK. FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
3112 020752 ;GROWING 0, AND SHIFTING 0.
3113 STARS
3114 ;*****
3115
3116 020752 012703 002564 BGNSEG MOV #BEGPAT,R3 ;GET START OF LIST
3117 020756 TRAP C$BSEG ;****START OF SEGMENT****
3118 020756 104404 1$:
3119 020760 CLR @RLBA ;CLEAR "BA"
3120 020760 005077 161216 MOV (R3),GDDAT ;SET EXPECTED
3121 020764 011337 002306 TST T.CNTRLR ;RL11
3122 020770 005737 002332 BEQ 3$ ;NO
3123 020774 001403 BIC #1,GDDAT ;BIT 0 CAN'T SET IN RLBA (UNIBUS)
3124 020776 042737 000001 002306 BIS (R3),@RLBA ;BIS RLBA WITH PATTERN
3125 021004 051377 161172 002310 MOV @RLBA,BDDAT ;READ "BA"
3126 021010 017737 161166 002310 CMP BDDAT,GDDAT ;DID RLBA LOAD PROPERLY?
3127 021016 023737 002310 002306 BEQ 2$ ;BRANCH IF YES
3128 021024 001404
3129
3130 021026 ERRDF 9,EM63,ERR2 ;WRONG DATA IN RLBA
3131 021026 104455 TRAP C$ERRDF
3132 021030 000011 .WORD 9
3133 021032 010475 .WORD EM63
3134 021034 011662 .WORD ERR2
3135 021036 2$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
3136 021036 104410 TRAP C$ESCAPE
3137 021040 000012 .WORD 10000$-.
3138
3139 021042 005723 TST (R3)+ ;GET NEXT PATTERN
3140 021044 020327 002772 CMP R3,#ENDPAT ;DID WE COMPLETE LIST
3141 021050 001343 BNE 1$ ;NO, GO BACK FOR NEXT.
3142 021052 ENDSEG ;****END OF SEGMENT****
3143 021052 10000$:
3144 021052 104405 TRAP C$ESEG

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 59
 (VRLAC.P11 31-AUG-82 11:25 **TEST 17** - BIS OF RLBA

SEQ 0059

```

3145 021054 FNDTST ;****END OF TEST****
3146 021054 L10043:
3147 021054 104401 TRAP C$ETST
3148
3149
3150 .SBTTL **TEST 18** - BIC OF RLBA
3151
3152 021056 BGNTST ;****START OF TEST****
3153
3154 021056 STARS
3155 ;*****
3156 ;TEST THAT THE 'BIC' INSTRUCTION WILL WORK ON THE BUS
3157 ;ADDRESS REGISTER. BITS 15-1 ARE TESTED WITH 4 PATTERNS
3158 ;GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0.
3159 021056 STARS
3160 ;*****
3161
3162
3163 021056 012703 002564 BGNSEG MOV #BEGPAT,R3 ;GET START OF LIST
3164 021062 TRAP C$BSEG ;****START OF SEGMENT****
3165 021062 104404
3166 021064
3167 021064 012777 177776 161110 1$: MOV #-2,@RLBA ;SET RLBA TO ALL 1'S (BIT 0=0)
3168 021072 012737 177776 002306 MOV #-2,GDDAT ;SET UP EXPECTED RESULTS
3169 021100 041337 002306 BIC (R3),GDDAT ;IN GDDAT
3170 021104 041377 161072 BIC (R3),@RLBA ;BIC RLBA
3171 021110 017737 161066 002310 MOV @RLBA,BDDAT ;READ RLBA
3172 021116 023737 002310 002306 CMP BDDAT,GDDAT ;BIC WORK OKAY?
3173 021124 001404 BEQ 2$ ;IF YES BRANCH
3174
3175 021126 ERRDF 10.,EM64,ERR2 ;WRONG DATA IN RLBA
3176 021126 104455 TRAP C$ERRDF
3177 021130 000012 .WORD 10
3178 021132 010556 .WORD EM64
3179 021134 011662 .WORD ERR2
3180 021136 2$: ESCAPE SEG ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
3181 021136 174410 TRAP C$ESCAPE
3182 021140 001012 .WORD 10000$-.
3183
3184 021142 005723 TST (R3)+ ;GET NEXT PATTERN
3185 021144 020327 002772 CMP R3,#ENDPAT ;HAVE WE COMPLETED LIST
3186 021150 001345 BNE 1$ ;NO, GO BACK FOR NEXT
3187 021152 ENDSEG ;****END OF SEGMENT****
3188 021152 10000$:
3189 021152 104405 TRAP C$ESEG
3190 021154 ENDTST ;****END OF TEST****
3191 021154 L10044:
3192 021154 104401 TRAP C$ETST
3193
3194
3195 .SBTTL **TEST 19** - BIS OF RLDA
3196
3197 021156 BGNTST ;****START OF TEST****
3198
3199 021156 STARS
3200 ;*****

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 60
 CVRLAC.P11 31-AUG-82 11:25 **TEST 19** - BIS OF RLDA

SEQ 0060

```

3201      ;TEST THAT THE 'BIS' INSTRUCTION WILL WORK ON THE DISK ADDRESS
3202      ;REGISTER. BITS 15-0 ARE TESTED WITH 4 PATTERNS, GROWING 1,
3203      ;SHIFTING 1, GROWING 0, AND SHIFTING 0.
3204      STARS
3205      ;*****
3206
3207
3208      021156 012703 002564      BGNSEG  MOV    #BEGPAT,R3      ;GET START OF LIST
3209      021162      TRAP    C$BSEG      ;*****START OF SEGMENT****
3210      021162 104404      1$:
3211      021164      CLR     @RLDA      ;CLEAR 'DA'
3212      021164 005077 161014      MOV     (R3),GDDAT      ;SET EXPECTED
3213      021170 011337 002306      BIS     (R3),@RLDA      ;BIS RLDA
3214      021174 051377 161004      MOV     @RLDA,BDDAT      ;READ RLDA
3215      021200 017737 161000 002310      CMP     BDDAT,GDDAT      ;IS RLDA CORRECT
3216      021206 023737 002310 002306      BEQ     2$      ;IF OKAY BRANCH
3217      021214 001404
3218
3219      021216      ERDF     11,EM65,ERR2      ;WRONG DATA IN RLDA
3220      021216 104455      TRAP    C$ERDF
3221      021220 000013      .WORD   11
3222      021222 010641      .WORD   EM65
3223      021224 011662      .WORD   ERR2
3224      021226      ESCAPE  SEG      ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
3225      021226 104410      TRAP    C$ESCAPE
3226      021230 000012      .WORD   10000$-.
3227
3228      021232 005723      TST     (R3)+      ;GET NEXT PATTERN
3229      021234 020327 002772      CMP     R3,#ENDPAT      ;HAVE WE FINISHED?
3230      021240 001351      BNE     1$      ;NO GO BACK
3231      021242      ENDSEG      ;*****END OF SEGMENT****
3232      021242 10060$:
3233      021242 104405      TRAP    C$ESEG
3234      021244      ENDTST
3235      021244      L10045:
3236      021244 104401      TRAP    C$ETST
3237
3238
3239      .SBTTL **TEST 20** - BIC OF RLDA
3240
3241      021246      BGNTEST      ;*****START OF TEST****
3242
3243      021246      STARS
3244      ;*****
3245      ;TEST THAT THE 'BIC' INSTRUCTION WORKS ON THE DISK
3246      ;ADDRESS REGISTER. ALL BITS ARE TESTED WITH FOUR
3247      ;PATTERNS: GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
3248      STARS
3249      ;*****
3250
3251
3252      021246 012703 002564      BGNSEG  MOV    #BEGPAT,R3      ;GET START OF LIST
3253      021252      TRAP    C$BSEG      ;*****START OF SEGMENT****
3254      021252 104404      1$:
3255      021254      MOV     #-1,@RLDA      ;SET RLDA TO ALL 1'S
3256      021254 012777 177777 160722

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 61
 CRAC.P11 31-AUG-82 11:25 **TEST 20** - BIC OF RLDA

SEQ 0061

```

3257 021262 012737 177777 002306      MOV    #-1,GDDAT      ;SET EXPECTED DATA
3258 021270 041337 002306      BIC     (R3),GDDAT    ;SET EXPECTED DATA
3259 021274 041377 160704      BIC     (R3),@RLDA    ;"BIC" RLDA
3260 021300 017737 160700 002310      MOV    @RLDA,BDDAT    ;READ RLDA
3261 021306 023737 002306 002310      CMP     GDDAT,BDDAT    ;DID "BIC" WORK?
3262 021314 001404      BEQ      2$      ;IF IT DID BRANCH
3263
3264 021316      ERRDF     12.,EM66,ERR2    ;WRONG DATA IN RLDA
3265 021316 104455      TRAP    C$ERDF
3266 021320 000014      .WORD    12
3267 021322 010722      .WORD    EM66
3268 021324 011662      .WORD    ERR2
3269 021326      2$:      ESCAPE    SEG      ;IF /FL:LOE SET LOOP, ELSE EXIT SEG
3270 021326 104410      TRAP    C$ESCAPE
3271 021330 000012      .WORD    10000$-
3272
3273 021332 005723      TST     (R3)+      ;GET NEXT PATTERN
3274 021334 020327 002772      CMP     R3,#ENDPAT    ;DONE?
3275 021340 001345      BNE      1$      ;NO GO BACK
3276 021342      ENDSEG      ;****END OF SEGMENT****
3277 021342 10000$:
3278 021342 104405      TRAP    C$ESEG
3279 021344      ENDTST
3280 021344 L10046:
3281 021344 104401      TRAP    C$ETST
3282
3283      .SBTTL **TEST 21** - BUS RESET OF RLCS
3284
3285      BGNIST      ;****START OF TEST****
3286 021346
3287
3288 021346      STARS
3289      ;*****
3290      ;TEST THAT A BUS RESET WILL CLEAR THE PROPER BITS
3291      ;OF THE CONTROL AND STATUS REGISTER. THOSE BITS ARE
3292      ;BITS 6-1,8,9,10,11,12,13,15. BIT 15 WILL CLEAR ONLY
3293      ;IF BIT 14 (DRIVE ERROR IS NOT SET). BIT 0 (DRIVE READY)
3294      ;IS A DON'T CARE. IF AT THE START UP THIS TEST BIT
3295      ;14 (DRIVE ERROR) IS SET WE WILL INSIST IF IS THERE AFTER
3296      ;THE "RESET" ALONG WITH BIT 15 (COMPOSITE ERROR). BITS
3297      ;15-10 ARE NOT WRITEABLE.
3298 021346      STARS
3299      ;*****
3300
3301
3302 021346      SETPRI    #PRI07      ;PRIORITY TO SEVEN
3303 021346 012700 000340      MOV     #PRI07,R0
3304 021352 104441      TRAP    C$SPRI
3305 021354 012777 000377 160616      MOV     #377,@RLCS    ;LOAD ALL RLCS LOADABLE BITS
3306 021362 012737 000200 002306      MOV     #CRDY,GDDAT    ;SETUP EXPECTED
3307 021370 032777 040000 160602      BIT     #DERR,@RLCS    ;DRIVE ERR SET?
3308 021376 001403      BEQ      1$      ;IF NOT DON'T EXPECT IT
3309 021400 052737 140000 002306      BIS     #DERR!ERR,GDDAT ;IT'S SET, INIT BETTER NOT CLR
3310 021406 012700 000100      1$:      MOV     #100,R0    ;SET UP A WAIT LOOP
3311 021412      BRESET
3312 021412 104433      TRAP    C$RESET

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 62
 CVRLAC.P11 31-AUG-82 11:25 **TEST 21** - BUS RESET OF RLCS

SEQ 0062

```

3313 021414 005300      2$: DEC      R0      ;WAIT IN CASE OF DRIVE ERROR
3314 021416 001376      BNE      2$      ;
3315 021420 017737 160554 0)2310      MOV      @RLCS,BDDAT ;READ RLCS
3316 021426 042737 000001 0)2310      BIC      #DRDY,BDDAT ;CLEAR OUT DRDY - DON'T CARE
3317 021434 023737 002310 502306      CMP      BDDAT,GDDAT ;DID INIT WORK
3318 021442 001404      BEQ      3$      ;YES, BRANCH
3319

```

```

3320 021444      ERRDF      13,EM67,EPR2 ;WRONG DATA IN RLCS
3321 021444 104455      TRAP      C$ERDF
3322 021446 000015      .WORD      13
3323 021450 011005      .WORD      EM67
3324 021452 011662      .WORD      ERR2

```

```

3325 021454      3$:
3326 021454      ENDTST      ;*****END OF TEST****
3327 021454      L10047:
3328 021454 104401      TRAP      C$ETST
3329

```

```

3330
3331      .SBTTL  **TEST 22** - BUS RESET OF RLBA
3332

```

```

3333 021456      BGNTST      ;*****START OF TEST****
3334

```

```

3335 021456      STARS
3336      ;*****
3337      ;TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
3338      ;BUS ADDRESS REGISTER. THE BUS ADDRESS IS LOADED WITH 177776
3339      ;AND IS EXPECTED TO BE ZERO AFTER THE RESET
3340 021456      STARS
3341      ;*****
3342

```

```

3343
3344 021456 012777 177776 160516      MOV      #-2,@RLBA ;SET BA TO ALL 1'S
3345 021464 005737 002332      TST      T,CNTRL ;RL11??
3346 021470 001403      BEQ      2$      ;NO
3347 021472 052777 000001 160502      BIS      #1,@RLBA
3348 021500 005037 002306      2$: CLR      GDDAT ;CLEAR EXPECTED DATA
3349 021504      BRESET ;ISSUE BUS INIT
3350 021504 104433      TRAP      C$RESET
3351 021506 017737 160470 002310      MOV      @RLBA,BDDAT ;READ RLBA
3352 021514 001404      BEQ      1$      ;IF CLEAR BRANCH
3353

```

```

3354 021516      ERRDF      14,EM70,ERR2 ;WRONG DATA IN RLBA
3355 021516 104455      TRAP      C$ERDF
3356 021520 000016      .WORD      14
3357 021522 011042      .WORD      EM70
3358 021524 011662      .WORD      ERR2

```

```

3359 021526      1$:
3360
3361 021526      ENDTST      ;*****END OF TEST****
3362 021526      L10050:
3363 021526 104401      TRAP      C$ETST
3364

```

```

3365
3366      .SBTTL  **TEST 23** - BUS RESET OF RLDA
3367

```

```

3368 021530      BGNTST      ;*****START OF TEST****

```

```

3369
3370 021530 STARS
3371 :*****
3372 :TEST THAT A BUS RESET WILL CLEAR THE ENTIRE
3373 :DISK ADDRESS REGISTER. THE DISK ADDRESS IS LOADED WITH 177777
3374 :AND IS EXPECTED TO BE ZERO AFTER THE RESET.
3375 021,30 STARS
3376 :*****
3377
3378
3379 021530 012777 177777 160446 MOV #-1,@RLDA ;SET DA TO ALL 1'S
3380 021536 005037 002306 CLR GDDAT ;CLEAR EXPECTED
3381 021542 BRESET ;ISSUE BUS INIT
3382 021542 104433 TRAP C$RESET
3383 021544 017737 160434 002310 MOV @RLDA,BDDAT ;READ RLDA
3384 021552 001404 BEQ 1$ ;IF CLEAR BRANCH
3385
3386 021554 ERRDF 15,EM71,ERR2 ;WRONG DATA IN RLDA
3387 021554 104455 TRAP C$ERDF
3388 021556 000017 .WORD 15
3389 021560 011077 .WORD EM71
3390 021562 011662 .WORD ERR2
3391 021564 1$:
3392
3393 021564 ENDTST ;****END OF TEST****
3394 021564 L10051:
3395 021564 104401 TRAP C$ETST
3396
3397
3398 .SBTTL **TEST 24** - UNIQUENESS OF RLCS
3399
3400 021566 BGNST ;****START OF TEST****
3401
3402 021566 STARS
3403 :*****
3404 :TEST THE UNIQUENESS OF THE CONTROL AND STATUS
3405 :REGISTER. THE RLBA AND RLDA ARE PRELOADED WITH
3406 :177776 AND 177777 RESPECTIVELY. THE RLCS IS THEN
3407 :LOADED TO INSURE THAT NEITHER THE RLBA OR RLDA
3408 :ARE MODIFIED BY THE WRITING OF THE RLCS.
3409 021566 STARS
3410 :*****
3411
3412
3413 021566 012737 000201 002252 MOV #DRDY!CRDY,LDCSR ;SET DRIVE AND CONTROLLER READY
3414 021574 012777 177776 160400 MOV #-2,@RLBA ;SET RLBA TO ALL 1'S
3415 021602 012777 177777 160374 MOV #-1,@RLDA ;SET RLDA TO ALL 1'S
3416 021610 013777 002252 160362 MOV LDCSR,@RLCS ;WRITE RLCS
3417
3418 ;CHECK THAT RLBA REMAINED UNEFFECTED
3419
3420 021616 022777 177776 160356 CMP #-2,@RLBA ;RLBA OKAY?
3421 021624 001412 BEQ 1$ ;YES, GO CHECK DA
3422
3423 021626 012737 177776 002306 MOV #-2,GDDAT ;SET UP EXPECTED
3424 021634 017737 160342 002310 MOV @RLBA,BDDAT ;READ RLBA

```

```

3425
3426 021642 ERRDF 16.,EM72,ERR2 ;CS MODIFIED BA
3427 021642 104455 TRAP C$ERDF
3428 021644 000020 .WORD 16
3429 021646 011134 .WORD EM72
3430 021650 011662 .WORD ERR2
3431 021652 1$: CKLOOP ;CHECK IF /FL:LOE IS SET
3432 021652 104406 TRAP C$CLP1
3433
3434 021654 022777 177777 160322 CMP #-1,@RLDA ;RLDA OKAY?
3435 021662 001412 BEQ 2$ ;YES, CONTINUE
3436
3437 021664 012737 177777 002306 MOV #-1,GDDAT ;SET UP EXPECTED
3438 021672 017737 160306 002310 MOV @RLDA,BDDAT ;READ DA
3439
3440 021700 ERRDF 17.,EM73,ERR2 ;CS MODIFIED DA
3441 021700 104455 TRAP C$ERDF
3442 021702 000021 .WORD 17
3443 021704 011167 .WORD EM73
3444 021706 011662 .WORD ERR2
3445 021710 2$:
3446
3447
3448 021710 ENDTST ;****END OF TEST****
3449 021710 L10052:
3450 021710 104401 TRAP C$ETST
3451
3452
3453 .SBTTL **TEST 25** - UNIQUENESS OF RLBA
3454
3455 021712 BGNIST ;****START OF TEST****
3456 021712 STARS
3457 ;*****
3458 ;TEST THE UNIQUENESS OF THE BUS ADDRESS REGISTER. THE
3459 ;RLCS AND RLDA ARE LOADED WITH XXX20X AND 177777
3460 ;RESPECTIVELY. THE RLBA IS THEN WRITTEN TO INSURE
3461 ;THAT NEITHER THE RLCS OR RLDA ARE MODIFIED
3462 ;BY WRITING THE RLBA.
3463 021712 STARS
3464 ;*****
3465
3466
3467 021712 012737 000200 002306 MOV #CRDY,GDDAT ;CONTROLLER READY
3468 021720 032777 040000 160252 BIT #DERR,@RLCS ;IF DRIVE ERROR IS
3469 021726 001403 BEQ 99$ ;SET THEN EXPECT IT
3470 021730 052737 140000 002306 BIS #ERR!DERR,GDDAT ;SET WHEN WE READ IT.
3471 021736 013777 002306 160234 99$: MOV GDDAT,@RLCS ;LOAD RLCS
3472 021744 012777 177777 160232 MOV #-1,@RLDA ;LOAD RLDA
3473 021752 005077 160224 CLR @RLBA ;CLEAR RLBA
3474
3475 ;CHECK IF RLCS IS OKAY
3476
3477 021756 017737 160216 002310 MOV @RLCS,BDDAT ;READ RLCS
3478 021764 042737 000001 002310 BIC #DRDY,BDDAT ;IGNORE DRIVE READY
3479 021772 023737 002310 002306 CMP BDDAT,GDDAT ;CS OK?
3480 022000 001404 BEQ 1$ ;YES, GO CHECK DA
  
```

```
3481
3482 022002 ERRDF 18,EM74,ERR2 :BA MODIFIED CS
3483 022002 104455 TRAP C$ERRDF
3484 022004 000022 .WORD 18
3485 022006 011222 .WORD EM74
3486 022010 011662 .WORD ERR2
3487 022012 1$: CKLOOP :CHECK IF /FL:LOE IS SET
3488 022012 104406 TRAP C$CLP1
3489
3490 022014 022777 177777 160162 CMP #-1,@RLDA :IS RLDA OKAY?
3491
3492 022022 001412 BEQ 2$ :IF OKAY BRANCH
3493
3494 022024 012737 177777 002306 MOV #-1,GDDAT :SET UP EXPECTED
3495 022032 017737 160146 002310 MOV @RLDA,BDDAT :READ RLDA
3496
3497 022040 ERRDF 19,EM75,ERR2 :BA MODIFIED DA
3498 022040 104455 TRAP C$ERRDF
3499 022042 000023 .WORD 19
3500 022044 011254 .WORD EM75
3501 022046 011662 .WORD ERR2
3502 022050 2$:
3503 022050 ENDTST :****END OF TEST****
3504 022050 L10053:
3505 022050 104401 TRAP C$ETST
3506
3507
3508 .SBTTL **TEST 26** - UNIQUENESS OF RLDA
3509
3510 022052 BGNTST :****START OF TEST****
3511
3512
3513 022052 STARS
3514 :*****
3515 :TEST THE UNIQUENESS OF THE DISK ADDRESS REGISTER. THE RLCS
3516 :AND RLBA ARE LOADED WITH XXX20X AND 177776
3517 :RESPECTIVELY. THE RLDA IS THEN WRITTEN TO INSURE
3518 :THAT NEITHER THE RLCS OR THE RLBA ARE MODIFIED
3519 :BY WRITING THE RLDA.
3520 022052 STARS
3521 :*****
3522
3523
3524 022052 012737 000200 002306 MOV #CRDY,GDDAT :CONTROLLER READY
3525 022060 032777 040000 160112 BIT #DERR,@RLCS :IF DRIVE ERROR SET
3526 022066 001403 BEQ 99$ :THEN EXPECT IT LATER
3527 022070 052737 140000 002306 BIS #ERR!DERR,GDDAT
3528 022076 013777 002306 160074 99$: MOV GDDAT,@RLCS :LOAD CS
3529 022104 012777 177776 160070 MOV #-2,@RLBA :LOAD BA WITH ALL 1'S
3530 022112 005077 160066 CLR @RLDA :CLEAR RLDA
3531
3532 :CHECK IF RLCS IS OKAY
3533
3534 022116 017737 160056 002310 MOV @RLCS,BDDAT :READ RLCS
3535 022124 042737 000001 002310 BIC #DRDY,BDDAT :IGNORE DRIVE READY
3536 022132 023737 002306 002310 CMP GDDAT,BDDAT :RLCS OKAY?
```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 66
 CVRLAC.P11 31-AUG-82 11:25 **TEST 26** - UNIQUENESS OF RLDA

SEQ 0066

```

3537 022140 001404      BEQ      1$      ;YES, THEN BRANCH
3538
3539 022142      ERRDF    20.,EM76,ERR2 ;DA MODIFIED CS
3540 022142 104455      TRAP    C$ERRDF
3541 022144 000024      .WORD    20
3542 022146 L 306      .WORD    EM76
3543 022150 011662      .WORD    ERR2
3544 022152      1$:      CKLOOP      ;CHECK IF /FL:LOE IS SET
3545 022152 104406      TRAP    C$CLP1
3546
3547 022154 022777 177776 160020      CMP      #-2,@RLBA      ;IS RLBA OKAY?
3548 022162 001412      BEQ      2$      ;BRANCH IF OKAY
3549
3550 022164 012737 177776 002306      MOV      #-2,GDDAT      ;SET UP EXPECTED
3551 022172 017737 160004 002310      MOV      @RLBA,BDDAT      ;READ RLBA
3552
3553 022200      ERRDF    21.,EM77,ERR2 ;DA MODIFIED BA
3554 022200 104455      TRAP    C$ERRDF
3555 022202 000025      .WORD    21
3556 022204 011341      .WORD    EM77
3557 022206 011662      .WORD    ERR2
3558 022210      2$:
3559
3560
3561 022210      ENDTST      ;****END OF TEST****
3562 022210      L10054:
3563 022210 104401      TRAP    C$ETST
3564
3565      .SBTTL  **TEST 27** - UNIQUENESS OF RLMP
3566
3567 022212      BGNTST      ;****START OF TEST****
3568
3569
3570 022212      STARS
3571      ;*****
3572      ;TEST THE UNIQUENESS OF THE MULTI-PURPOSE REGISTER
3573      ;WE WILL WRITE THE RLCS, RLBA, AND THE RLDA, THEN THE
3574      ;RLMP IS WRITTEN. WE THEN GO BACK AN VERIFY THE CONTENTS
3575      ;OF THE RLCS, RLBA, RLDA.
3576 022212      STARS
3577      ;*****
3578
3579
3580 022212 012737 000200 002306      MOV      #CRDY,GDDAT      ;CONTROLLER READY
3581 022220 032777 040000 157752      BIT      #DERR,@RLCS      ;IF DRIVE ERROR SET
3582 022226 001403      BEQ      99$      ;THE EXPECT IT LATER
3583 022230 052737 140000 002306      BIS      #ERR!DERR,GDDAT
3584 022236 013777 002306 157734      99$:      MOV      GDDAT,@RLCS      ;LOAD CS
3585 022244 012777 177776 157730      MOV      #-2,@RLBA      ;LOAD BA WITH ALL 1'S
3586 022252 012777 177777 157724      MOV      #-1,@RLDA      ;LOAD RLDA
3587 022260 005077 157722      CLR      @RLMP      ;WRITE RLMP
3588
3589      ;CHECK IF RLCS IS OKAY
3590
3591 022264 017737 157710 002310      MOV      @RLCS,BDDAT      ;READ RLCS
3592 022272 042737 000001 002310      BIC      #DRDY,BDDAT      ;IGNORE DRIVE READY

```

```
3593 022300 023737 002306 002310      CMP      GDDAT,BDDAT      ;RLCS OKAY?
3594 022306 C01404                      BEQ      1$              ;YES, THEN BRANCH
3595
3596 022310                      ERRDF      22.,EM44,ERR2      ;MP MODIFIED CS
3597 022310 104455                      TRAP      C$ERDF
3598 022312 000026                      .WORD      22
3599 022314 010210                      .WORD      EM44
3600 022316 011662                      .WORD      ERR2
3601 022320                      1$:      CKLOOP
3602 022320 104406                      TRAP      C$CLP1      ;CHECK IF /FL:LOE IS SET
3603
3604 022322 022777 177776 157652      CMP      #-2,@RLBA      ;IS RLBA OKAY?
3605 022330 001412                      BEQ      2$              ;BRANCH IF OKAY
3606
3607 022332 012737 177776 002306      MOV      #-2,GDDAT      ;SET UP EXPECTED
3608 022340 017737 157636 002310      MOV      @RLBA,BDDAT    ;READ RLBA
3609
3610 022346                      ERRDF      23.,EM45,ERR2      ;MP MODIFIED BA
3611 022346 104455                      TRAP      C$ERDF
3612 022350 000027                      .WORD      23
3613 022352 010243                      .WORD      EM45
3614 022354 011662                      .WORD      ERR2
3615 022356                      2$:      CKLOOP
3616 022356 104406                      TRAP      C$CLP1      ;CHECK IF /FL:LOE IS SET
3617 022360 022777 177777 157616      CMP      #-1,@RLDA      ;DISK ADDRESS OKAY
3618 022366 001412                      BEQ      3$              ;YES, CONTINUE
3619
3620 022370 017737 157610 002310      MOV      @RLDA,BDDAT    ;SET UP BAD
3621 022376 012737 177777 002306      MOV      #-1,GDDAT      ;SET UP EXPECTED
3622
3623 022404                      ERRDF      24.,EM46,ERR2      ;MP MODIFIED DA
3624 022404 104455                      TRAP      C$ERDF
3625 022406 000030                      .WORD      24
3626 022410 010276                      .WORD      EM46
3627 022412 011662                      .WORD      ERR2
3628
3629 022414                      3$:
3630
3631
3632 022414                      ENDTST
3633 022414 L10055:
3634 022414 104401                      TRAP      C$ETST
3635
3636
3637
3638
3639
```

.SBTTL **TEST 28** - RLV11 MAINT. FORCED OPI TEST,LESS THAN 510 WORDS

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

STARS

;PERFORM RLV11 MAINTENANCE FUNCTION 0 WITH LESS THAN 510 WORDS
;TO FORCE OPI ERROR. THE TEST SHOULD FORCE COMPOSITE ERROR(BIT 15),
;HEADER NOT FOUND(BIT12) AND OPI(BIT 10). DRIVE ERROR IS IGNORED.
STARS

3648 022416

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 68
 (VRLAC.P11 31-AUG-82 11:25

TEST 28 - RLV11 MAINT. FORCED OPI TEST, LESS THAN 510 WORDS

SEQ 0068

```

3649          022416 005737 002332          ;*****
3650          022422 001040          TST      T.CNTRL          ;RLV11?
3651          022424 012703 002774          BNE      10$          ;NO,EXIT TEST
3652          022430 012704 003066          1$:      MOV      #PATCRC,R3          ;GET CRC PATTERN TABLE
3653          022434 011337 022450          MOV      #PATDAT,R4          ;GET DATA PATTERN TABLE
3654          022440 004537 015514          MOV      (R3),2$          ;STORE CRC PATTERN
3655          022444 000000 015514          MOV      (R4),3$          ;STORE DATA PATTERN
3656          022450 000000 015752          JSR      R5,CALCRC          ;CALCULATE CRC BEFORE TEST
3657          022452 000000          2$:      .WORD      0
3658          022456 000000          JSR      R5,SETPAT          ;SETUP PATTERN BEFORE TEST
3659          022460 004537 016114          3$:      .WORD      0
3660          022462 000000          BGNSEG
3661          022466 000000          TRAP     C$BSEG
3662          022470 006324 016724          JSR      R5,LDFUN          ;PERFORM MAINT FUNCTION
3663          022472 004537 015324          MAINT    -507          ;LESS THAN 510 WORDS
3664          022474 000404          MATMES   ;MAINT. MESSAGE
3665          022500 004537 015324          JSR      R5,WTCRDY
3666          022502 000404          CKLOOP   ;LOOP SWITCH
3667          022506 004537 015324          TRAP     C$CLP1
3668          022510 000404          JSR      R5,CHKOP1          ;CHECK FOR EXPECTED ERRORS
3669          022512 000404          BR       4$          ;EXPECTED ERRORS FOUND,EXIT TEST
3670          022514 000404          ERRDF    30,EM27,ERR10
3671          022516 000404          TRAP     C$ERDF
3672          022520 000404          .WORD     30
3673          022522 000404          .WORD     EM27
3674          022524 000404          .WORD     ERR10
3675          022526 000404          4$:      CKLOOP
3676          022528 000404          TRAP     C$CLP1
3677          022530 000404          ENDSEG
3678          022532 000404          10000$:  TRAP     C$ESEG
3679          022534 000404          10$:
3680          022536 000404          ENDTST
3681          022538 000404          L10056:  TRAP     C$ETST
3682          022540 000404          .SBTTL   **TEST 29** - RLV11 MAINT. FORCED OPI TEST,MORE THAN 511 WORDS
3683          022542 000404          BGNTEST   ;*****START OF TEST*****
3684          022544 000404          STARS
3685          022546 000404          ;*****
3686          022548 000404          ;PERFORM RLV11 MAINTENANCE FUNCTION 0 WITH MORE THAN 511 WORDS
3687          022550 000404          ;TO FORCE OPI ERROR. THE TEST SHOULD FORCE COMPOSITE ERROR(BIT 15).
3688          022552 000404          ;HEADER NOT FOUND(BIT12) AND OPI(BIT 10). DRIVE ERROR IS IGNORED.
3689          022554 000404          STARS
3690          022556 000404          ;*****
3691          022558 000404          TST      T.CNTRL          ;RLV11?
3692          022560 000404          BNE      10$          ;NO,EXIT TEST
3693          022562 000404          1$:      MOV      #PATCRC,R3          ;GET CRC PATTERN TABLE
3694          022564 000404          MOV      #PATDAT,R4          ;GET DATA PATTERN TABLE
3695          022566 000404          MOV      (R3),2$          ;STORE CRC PATTERN
3696          022568 000404          MOV      (R4),3$          ;STORE DATA PATTERN
3697          022570 000404          JSR      R5,CALCRC          ;CALCULATE CRC BEFORE TEST
3698          022572 000404
3699          022574 000404
3700          022576 000404
3701          022578 000404
3702          022580 000404
3703          022582 000404
3704          022584 000404

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 69
 CVRLAC.P11 31-AUG-82 11:25

TEST 29 - RLV11 MAINT. FORCED OPI TEST, MORE THAN 511 WORDS

SEQ 0069

3705	022560	000000		2\$:	.WORD	0	
3706	022562	004537	015752		JSR	R5,SETPAT	:SETUP PATTERN BEFORE TEST
3707	022566	000000		3\$:	.WORD	0	
3708	022570				BGNSEG		
3709	022570	104404			TRAP	C\$BSEG	
3710	022572	004537	016114		JSR	R5,LDFUN	:PERFORM MAINT FUNCTION
3711	022576	000000			MAINT		
3712	022600	177266			-512		:MORE THAN 511 WORDS
3713	022602	006324			MATMES		:MAINT. MESSAGE
3714	022604	004537	016724		JSR	R5,WTCRDY	
3715	022610				CKLOOP		:LOOP SWITCH
3716	022610	104406			TRAP	C\$CLP1	
3717	022612	004537	015324		JSR	R5,CHKOPI	:CHECK FOR EXPECTED ERRORS
3718	022616	000404			BR	4\$	:EXPECTED ERRORS FOUND,EXIT TEST
3719	022620				ERRDF	31,EM30,ERR10	
3720	022620	104455			TRAP	C\$ERDF	
3721	022622	000037			.WORD	31	
3722	022624	010113			.WORD	EM30	
3723	022626	012206			.WORD	ERR10	
3724	022630			4\$:	CKLOOP		
3725	022630	104406			TRAP	C\$CLP1	
3726	022632				ENDSEG		
3727	022632			10000\$:			
3728	022632	104405			TRAP	C\$ESEG	
3729	022634			10\$:			
3730							
3731	022634			ENDTST			
3732	022634			L10057:			
3733	022634	104401			TRAP	C\$ETST	
3734							
3735				.SBTTL	**TEST 30** - RLV11 MAINT. FORCED OPI TEST - INTERRUPT MODE		
3736				BGNTST		:****START OF TEST****	
3737	022636						
3738				STARS			
3739	022636				:*****		
3740					:PERFORM TEST OF INTERRUPT BY ISSUING RLV11 MAINTENANCE FUNCTION 0		
3741					:WITH MORE THAN 511 WORDS TO FORCE OPI ERROR. CHECK INTERRUPT		
3742					:OPERATION AND REPORT IF ERROR OCCURS.		
3743					STARS		
3744	022636				:*****		
3745							
3746							
3747	022636	005737	002332		TST	T.CNTRL	:RLV11?
3748	022642	001052			BNE	10\$	:NO,EXIT TEST
3749	022644	012703	002774	1\$:	MOV	#PATCRC,R3	:GET CRC PATTERN TABLE
3750	022650	012704	003066		MOV	#PATDAT,R4	:GET DATA PATTERN TABLE
3751	022654	011337	022670		MOV	(R3),2\$	:STORE CRC PATTERN
3752	022660	011437	022676		MOV	(R4),3\$	:STORE DATA PATTERN
3753	022664	004537	015514		JSR	R5,CALCRC	:CALCULATE CRC
3754	022670	000000		2\$:	.WORD	0	
3755	022672	004537	015752		JSR	R5,SETPAT	:SETUP PATTERN
3756	022676	000000		3\$:	.WORD	0	
3757	022700				BGNSEG		
3758	022700	104404			TRAP	C\$BSEG	
3759	022702				SETPRI	#PRI00	:SET PRIORITY TO ZERO
3760	022702	012700	000000		MOV	#PRI00,R0	

MAIN. MACV11 30(1046) 31-AUG-82 11:50 PAGE 70
 CVRLAC.P11 31-AUG-82 11:25 **TEST 30** - RLV11 MAINT. FORCED OPI TEST - INTERRUPT MODE

SEQ 0070

3761	022706	104441		TRAP	C\$SPRI	
3762	022710	005037	002250	CLR	INTFLG	:CLEAR INT. FLAG
3763	022714	004537	016114	JSR	R5, LDFUN	
3764	022720	000100		MAINT:INTEN		:MAINT FUNCTION, INT DRIVEN
3765	022722	177266		-512		:MORE THAN 511 TO FORCE OPI ERROR
3766	022724	006364		MATINT		
3767	022726	004537	016724	JSR	R5, WTCRDY	:WAIT FOR READY
3768	022732			CKLOOP		
3769	022732	104406		TRAP	C\$CLP1	
3770	022734			SETPRI	#PRI07	
3771	022734	012700	000340	MOV	#PRI07, R0	
3772	022740	104441		TRAP	C\$SPRI	
3773	022742	005737	002250	TST	INTFLG	:CHECK IF INTERRUPT OCCURRED
3774	022746	001004		BNE	4\$	
3775	022750			ERRDF	32, EM24, ERRO	
3776	022750	104455		TRAP	C\$ERDF	
3777	022752	000040		.WORD	32	
3778	022754	007664		.WORD	EM24	
3779	022756	011632		.WORD	ERRO	
3780	022760	005037	002250	4\$: CLR	INTFLG	:CLEAR INT. FLAG
3781	022764			CKLOOP		
3782	022764	104406		TRAP	C\$CLP1	
3783	022766			ENDSEG		
3784	022766			10000\$:		
3785	022766	104405		TRAP	C\$ESEG	
3786	022770			10\$:		
3787				ENDTST		
3788	022770			L10060:		
3789	022770			TRAP	C\$ETST	
3790	022770	104401				
3791						
3792						
3793				.SBTTL	**TEST 31** - RLV11 OPI TIMEOUT TEST	
3794						
3795	022772			BGNIST		:START OF TEST
3796						
3797	022772			STARS		
3798				*****		
3799				:PERFORM RLV11 MAINTENANCE FUNCTION (0) WITH INTERRUPT MODE. FORCE		
3800				:OPI TIMEOUT BY SETTING WORD COUNT TO -512 WORDS. MEASURE OPI TIMEOUT		
3801				:AND COMPARE TO MIN. AND MAX. LIMITS.		
3802	022772			STARS		
3803				*****		
3804						
3805	022772	005737	002332	TST	T.CNTRL	:RLV11?
3806	022776	001402		BEQ	1\$	:YES, PERFORM TEST
3807	023000	000137	023322	JMP	10\$	:RLV11, EXIT TEST
3808	023004	012703	002774	1\$: MOV	#PATCRC, R3	:GET CRC PATTERN TABLE
3809	023010	012704	003066	MOV	#PATDAT, R4	:GET DATA PATTERN TABLE
3810	023014	011337	023030	MOV	(R3), 2\$	:STORE CRC PATTERN
3811	023020	011437	023036	MOV	(R4), 3\$	:STORE DATA PATTERN
3812	023024	004537	015514	JSR	R5, CALCRC	:CALCULATE CRC BEFORE TEST
3813	023030	000000		2\$: .WORD	0	
3814	023032	004537	015752	JSR	R5, SETPAT	:SETUP PATTERN BEFORE TEST
3815	023036	000000		3\$: .WORD	0	
3816	023040			BGNSEG		

3817	023040	104404			TRAP	C\$BSEG	
3818	023042				CLRVEC	BVEC	:CLEAR PRESENT INT. VECTOR
3819	023042	013700	002214		MOV	BVEC,R0	
3820	023046	104436			TRAP	C\$VEC	
3821	023050				SETVEC	BVEC,#TIMSRV,#340	
3822	023050	012746	000340		MOV	#340,-(SP)	
3823	023054	012746	016652		MOV	#TIMSRV,-(SP)	
3824	023060	013746	002214		MOV	BVEC,-(SP)	
3825	023064	012746	000003		MOV	#3,-(SP)	
3826	023070	104437			TRAP	C\$SVEC	
3827	023072	062706	000010		ADD	#10,SP	
3828	023076	013737	002340	002310	MOV	OPIMX,BDDAT	:TAKE MAX LIMIT AND
3829	023104	005002			CLR	R2	:DIVIDE BY 10
3830	023106	162737	000012	002310	SUB	#10.,BDDAT	:RESULT IN R2
3831	023114	100402			BMI	201\$	:DONE DIVIDE?
3832	023116	005202			INC	R2	
3833	023120	000772			BR	200\$	:NEXT DIVIDE
3834	023122			201\$:	SETPRI	#PRI00	:SETUP FOR WAIT ABORT
3835	023122	012700	000000		MOV	#PRI00,R0	
3836	023126	104441			TRAP	C\$SPRI	
3837	023130	005037	002250		CLR	INTFLG	:CLEAR INTERRUPT FLAG
3838	023134	005000			CLR	R0	:OPI COUNTER
3839	023136	004537	016114		JSR	R5,LDFUN	:PERFORM MAINT. FUNCTION
3840	023142	000100			MAINT:INTEN		:MAINT FUNCTION WITH INT.MODE
3841	023144	177266			-512		:WORD COUNT
3842	023146	006364			MATINT		:MAINT MESSAGE
3843	023150	004537	016032		JSR	R5,WDELAY	:DELAY
3844	023154	000012		100\$:	10.		:APPROX 10 MSECS
3845	023156	062700	000012		ADD	#10.,R0	:10 MSEC DELAY COUNTER
3846	023162	005737	002250		TST	INTFLG	:TEST INTERRUPT FLAG
3847	023166	001002			BNE	110\$	:CHECK TIMER IF INTFLG = 1
3848	023170	005302			DEC	R2	:DONE WITH MAX DELAY?
3849	023172	001366			BNE	100\$	:DO ANOTHER DELAY
3850	023174	010037	002310		MOV	R0,BDDAT	:STORE RESULT
3851	023200	005737	002250		TST	INTFLG	:CHECK INT. FLG
3852	023204	001004			BNE	4\$	
3853	023206				ERRDF	33.,EM24,ERR0	:ERROR ON INTERRUPT
3854	023206	104455			TRAP	C\$ERDF	
3855	023210	000041			.WORD	33	
3856	023212	007664			.WORD	EM24	
3857	023214	011632			.WORD	ERR0	
3858	023216	005037	002250		CLR	INTFLG	
3859	023222				CKLOOP		
3860	023222	104406			TRAP	C\$CLP1	
3861							
3862							:CHECK THAT OPI IS WITHIN LIMITS
3863							
3864	023224			7\$:	SETPRI	#PRI07	
3865	023224	012700	000340		MOV	#PRI07,R0	
3866	023230	104441			TRAP	C\$SPRI	
3867	023232	023737	002340	002310	CMP	OPIMX,BDDAT	:IS OPI WITHIN LIMITS
3868	023240	002404			BLT	8\$	:NO,REPORT ERROR
3869	023242	023737	002336	002310	CMP	OPIMN,BDDAT	:WITHIN LIMITS?
3870	023250	003404			BLE	9\$	:YES
3871	023252			8\$:	ERRDF	34.,EM31,ERR11	:OPI TIMING INCORRECT
3872	023252	104455			TRAP	C\$ERDF	

```
3873 023254 000042      .WORD 34
3874 023256 010167      .WORD EM31
3875 023260 012250      .WORD ERR11
3876 023262      9$: CKLOOP
3877 023262 104406      TRAP C$CLP1
3878 023264      CLRVEC BVEC      ;CLEAR PRESENT VECTOR AND RESET OLD
3879 023264 013700 002214 MOV BVEC,RO
3880 023270 104436      TRAP C$CVEC
3881 023272      SETVEC BVEC,#INTSRV,#340
3882 023272 012746 000340 MOV #340,-(SP)
3883 023276 012746 016644 MOV #INTSRV,-(SP)
3884 023302 013746 002214 MOV BVEC,-(SP)
3885 023306 012746 000003 MOV #3,-(SP)
3886 023312 104437      TRAP C$SVEC
3887 023314 062706 000010 ADD #10,SP
3888 023320      ENDSEG
3889 023320      10000$: TRAP C$ESEG
3890 023320 104405      10$:
3891 023322
3892
3893 023322      ENDTST
3894 023322      L10061:
3895 023322 104401      TRAP C$ETST
3896
3897
3898      .SBTTL **TEST 32** - TEST RLV11 MAINT. FUNCTION - FLAG MODE
3899
3900 023324      BGNST      ;*****START OF TEST*****
3901
3902 023324      STARS
3903      ;*****
3904      ;PERFORM RLV11 MAINTENANCE FUNCTION 0 IN FLAG MODE AND CHECK
3905      ;FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
3906      ;WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER
3907      ;THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
3908      ;RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
3909      ;FIFO INTO BUF2 MEMORY FOR PROPER DATA.
3910      ;CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
3911      ;A VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.
3912 023324      STARS
3913      ;*****
3914 023324 005737 002332      TST T.CNTRL      ;RLV11?
3915 023330 001402      BEQ 100$      ;YES,RLV11
3916 023332 000137 024134      JMP 10$      ;NO,SKIP TEST
3917 023336 012703 002774      100$: MOV #PATCRC,R3      ;GET CRC PATTERN TABLE
3918 023342 012737 003066 002352      MOV #PATDAT,PATSAV      ;GET DATA PATTERN TABLE
3919 023350 011337 023366      101$: MOV (R3),102$      ;STORE CRC PATTERN FOR CALCULATION
3920 023354 017737 156772 023376      MOV @PATSAV,103$      ;STORE DATA PAT. FOR BUFFER FILL
3921 023362 004537 015514      JSR R5,CALCRC      ;CALCULATE CRC BEFORE TEST
3922 023366 000000      102$: .WORD 0      ;PATTERN FOR CRC TEST
3923 023370      BGNSEG
3924 023370 104404      TRAP C$BSEG
3925 023372 004537 015752      JSR R5,SETPAT      ;SETUP PATTERN IN BUFFER
3926 023376 000000      103$: .WORD 0      ;BUFFER PATTERN
3927 023400 004537 016114      JSR R5,LDFUN      ;PERFORM MAINT. FUNCTION
3928 023404 000000      MAINT      ;MAINT FUNCTION FLAG DRIVEN
```

3929	023406	177001			-511.		:WORD COUNT
3930	023410	006324			MATMES		:MESSAGE
3931	023412	004537	016724		JSR	R5,WTCRDY	:WAIT FOR READY
3932	023416				CKLOOP		
3933	023416	104406			TRAP	C\$CLP1	
3934	023420	004537	014600		JSR	R5,CHERR	:CHECK CONTROLLER FOR ERRORS
3935	023424				CKLOOP		
3936	023424	104406			TRAP	C\$CLP1	
3937	023426	012737	005756	002306	MOV	#BUF1+1776,GDDAT	
3938	023434	013737	002234	002310	MOV	E.BA,BDDAT	
3939	023442	023737	002306	002310	CMP	GDDAT,BDDAT	:TEST BA REGISTER
3940	023450	001404			BEQ	1\$	
3941	023452				ERRDF	35.,EM10,ERR4	:DATA WRONG IN BA REGISTER
3942	023452	104455			TRAP	C\$ERDF	
3943	023454	000043			.WORD	35	
3944	023456	007046			.WORD	EM10	
3945	023460	012026			.WORD	ERR4	
3946	023462				1\$: CKLOOP		:CHECK FOR LOOP MODE
3947	023462	104406			TRAP	C\$CLP1	
3948	023464	013737	002224	002306	MOV	B.DA,GDDAT	:GET BEFORE DA REGISTER
3949	023472	013737	002236	002310	MOV	E.DA,BDDAT	
3950	023500	005037	002274		CLR	TEMP1	
3951	023504	113737	002224	002274	MOVB	B.DA,TEMP1	
3952	023512	062737	000006	002274	ADD	#6,TEMP1	:+6 TO DA LOW BYTE
3953	023520	113737	002274	002306	MOVB	TEMP1,GDDAT	:STORE LOW BYTE OF DA
3954	023526	023737	002306	002310	CMP	GDDAT,BDDAT	
3955	023534	001404			BEQ	2\$	
3956	023536				ERRDF	36.,EM12,ERR4	
3957	023536	104455			TRAP	C\$ERDF	
3958	023540	000044			.WORD	36	
3959	023542	007150			.WORD	EM12	
3960	023544	012026			.WORD	ERR4	
3961	023546				2\$: CKLOOP		
3962	023546	104406			TRAP	C\$CLP1	
3963	023550	013737	002314	002306	MOV	GDCRCA,GDDAT	:GET CRC OF DA+3 VALUE
3964	023556	013737	002240	002310	MOV	E.MP,BDDAT	:GET CONTROLLER CRC OF DA+3
3965	023564	023737	002306	002310	CMP	GDDAT,BDDAT	
3966	023572	001404			BEQ	3\$	
3967	023574				ERRDF	37.,EM20,ERR4	
3968	023574	104455			TRAP	C\$ERDF	
3969	023576	000045			.WORD	37	
3970	023600	007400			.WORD	EM20	
3971	023602	012026			.WORD	ERR4	
3972	023604				3\$: CKLOOP		
3973	023604	104406			TRAP	C\$CLP1	
3974	023606	013737	002316	002306	MOV	GDCRCB,GDDAT	
3975	023614	013737	002242	002310	MOV	E.MP1,BDDAT	
3976	023622	023737	002306	002310	CMP	GDDAT,BDDAT	
3977	023630	001404			BEQ	4\$	
3978	023632				ERRDF	38.,EM21,ERR4	
3979	023632	104455			TRAP	C\$ERDF	
3980	023634	000046			.WORD	38	
3981	023636	007453			.WORD	EM21	
3982	023640	012026			.WORD	ERR4	
3983	023642				4\$: CKLOOP		
3984	023642	104406			TRAP	C\$CLP1	

MAIN. MACV11 30(1046) 31-AUG-82 11:50 PAGE 74
 CVRLAC.P11 31-AUG-82 11:25 **TEST 32** - TEST RLV11 MAINT. FUNCTION - FLAG MODE

SEQ 0074

3985	023644	005037	002360	CLR	SAVCNT	:CLEAR BAD WORD COUNTER
3986	023650	005037	002304	CLR	CHECK	:CLEAR PRINT HEADER INDICATOR
3987	023654	012704	003760	MOV	#BUF1,R4	:GOOD DATA STORED IN BUF1
3988	023660	012702	004760	MOV	#BUF2,R2	:DATA BUFFER WRITTEN INTO BY MAINT.
3989	023664	012701	000377	MOV	#255,R1	
3990	023670	011437	002306	5\$: MOV	(R4),GDDAT	:EXPECTED DATA
3991	023674	011237	002310	MOV	(R2),BDDAT	:GET DATA FROM BUFFER
3992	023700	023737	002306	002310 CMP	GDDAT,BDDAT	
3993	023706	001440		BEQ	7\$	:DATA COMPARE
3994	023710	010237	002276	MOV	R2,TMP0	:DATA ERR-GET ADDRESS
3995	023714	005237	002360	INC	SAVCNT	:INC BAD WORD COUNTER
3996	023720	005737	002304	TST	CHECK	:CHECK IF FIRST TIME
3997	023724	001007		BNE	6\$	
3998	023726			ERRDF	39.,EM22,ERR3	
3999	023726	104455		TRAP	C\$ERDF	
4000	023730	000047		.WORD	39	
4001	023732	007535		.WORD	EM22	
4002	023734	011724		.WORD	ERR3	
4003	023736	005237	002304	INC	CHECK	:PRINT HEADER ONCE
4004	023742	000422		BR	7\$	
4005	023744			6\$: PRINTX	#FRMT14,E.BA,E.DA,TMP0,GDDAT,BDDAT	
4006	023744	013746	002310	MOV	BDDAT,-(SP)	
4007	023750	013746	002306	MOV	GDDAT,-(SP)	
4008	023754	013746	002276	MOV	TMP0,-(SP)	
4009	023760	013746	002236	MOV	E.DA,-(SP)	
4010	023764	013746	002234	MOV	E.BA,-(SP)	
4011	023770	012746	013477	MOV	#FRMT14,-(SP)	
4012	023774	012746	000006	MOV	#6,-(SP)	
4013	024000	010600		MOV	SP,R0	
4014	024002	104415		TRAP	C\$PNTX	
4015	024004	062706	000016	ADD	#16,SP	
4016	024010			7\$: CKLOOP		
4017	024010	104406		TRAP	C\$CLP1	
4018	024012	005722		TST	(R2)+	:INCREMENT BUFFER
4019	024014	005724		TST	(R4)+	:INCREMENT BUFFER
4020	024016	005301		DEC	R1	:FINISHED BUFFER?
4021	024020	001323		BNE	5\$	:RETURN FOR NEXT COMPARE
4022	024022	005737	002304	TST	CHECK	:CHECK FOR ERROR HEADER FLAG
4023	024026	001412		BEQ	77\$	
4024	024030			PRINTB	#FRMT98,SAVCNT	:PRINT BAD WORD COUNT
4025	024030	013746	002360	MOV	SAVCNT,-(SP)	
4026	024034	012746	013052	MOV	#FRMT98,-(SP)	
4027	024040	012746	000002	MOV	#2,-(SP)	
4028	024044	010600		MOV	SP,R0	
4029	024046	104414		TRAP	C\$PNTB	
4030	024050	062706	000006	ADD	#6,SP	
4031	024054	012737	123456	002306 77\$: MOV	#123456,GDDAT	:EXPECTED DATA IN LAST WORD+1
4032	024062	011237	002310	MOV	(R2),BDDAT	:GET LAST WORD+1 FROM BUF2
4033	024066	023737	002306	002310 CMP	GDDAT,BDDAT	
4034	024074	001404		BEQ	8\$	
4035	024076			ERRDF	40.,EM23,ERR4	
4036	024076	104455		TRAP	C\$ERDF	
4037	024100	000050		.WORD	40	
4038	024102	007624		.WORD	EM23	
4039	024104	012026		.WORD	ERR4	
4040	024106			8\$: CKLOOP		

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 75
 CVRLAC.P11 31-AUG-82 11:25 **TEST 32** - TEST RLV11 MAINT. FUNCTION - FLAG MODE

SEQ 0075

```

4041 024106 104406          TRAP      C$CLP1
4042 024110          ENDSEG
4043 024110          10000$:
4044 024110 104405          TRAP      C$ESEG
4045 024112 005723          TST       (R3)+          :INC CRC PATTERN
4046 024114 062737 000002 002352      ADD      #2,PATSAV          :UPDATE PATTERN TABLE
4047 024122 020327 003064          CMP      R3,#CRCEND          :CHECK FOR END
4048 024126 001402          BEQ       10$          :END OF TEST
4049 024130 000137 023350          JMP      101$          :CONTINUE TEST
4050
4051 024134          10$:
4052
4053 024134          ENDTST
4054 024134          L10062:
4055 024134 104401          TRAP      C$ETST
4056
4057
4058          .SBTTL  **TEST 33** - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE
4059
4060 024136          BGNIST          :****START OF TEST****
4061
4062 024136          STARS
4063          :*****
4064          :PERFORM RLV11 MAINTENANCE FUNCTION 0 (INT. MODE) AND CHECK
4065          :FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
4066          :WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER
4067          :THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
4068          :RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
4069          :FIFO INTO BUF2 MEMORY FOR PROPER DATA.
4070          :CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
4071          :VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.
4072 024136          STARS
4073          :*****
4074 024136 005737 002332          TST      T.CNTRL          :RLV11?
4075 024142 001402          BEQ       100$          :YES,RLV11
4076 024144 000137 025012          JMP      10$          :NO,SKIP TEST
4077 024150 012703 002774          100$: MOV      #PATCRC,R3          :GET CRC PATTERN
4078 024154 012737 003066 002352      MOV      #PATDAT,PATSAV :GET DATA PATTERN
4079 024162 011337 024200          101$: MOV      (R3),102$
4080 024166 017737 156160 024210      MOV      @PATSAV,103$
4081 024174 004537 015514          JSR      R5,CALCRC          :CALCULATE CRC BEFORE TEST
4082 024200 000000          102$: .WORD      0          :PATTERN FOR CRC TEST
4083 024202
4084 024202 104404          BGNSEG
4085 024204 004537 015752          TRAP      C$BSEG
4086 024210 000000          103$: JSR      R5,SETPAT          :SETUP PATTERN IN BUFFER
4087 024212          .WORD      0          :BUFFER PATTERN
4088 024212 012700 000000          SETPRI  #PRI00          :SET PRIORITY TO ZERO
4089 024216 104441          MOV      #PRI00,R0
4090 024220 005037 002250          TRAP      C$SPRI
4091 024224 004537 016114          CLR      INTFLG          :CLEAR INT. FLAG
4092 024230 000100          JSR      R5,LDFUN          :PERFORM MAINT. FUNCTION
4093 024232 177001          MAINT!INTEN          :MAINT FUNCTION INT. DRIVEN
4094 024234 006364          -511.          :WORD COUNT
4095 024236 004537 016724          MAT!NT          :MESSAGE
4096 024242          JSR      R5,WTCRDY          :WAIT FOR READY
          CKLOOP

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 76
 (VRLAC.P11 31-AUG-82 11:25 **TEST 33** - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

SEQ 0076

4097	024242	104406			TRAP	C\$CLP1	
4098	024244				SETPRI	#PRI07	
4099	024244	012700	000340		MOV	#PRI07,R0	
4100	024250	104441			TRAP	C\$SPRI	
4101	024252	005737	002250		TST	INTFLG	
4102	024256	001004			BNE	104\$	
4103	024260				ERRDF	41.,EM24,ERR0	
4104	024260	104455			TRAP	C\$ERDF	
4105	024262	000051			.WORD	41	
4106	024264	007664			.WORD	EM24	
4107	024266	011632			.WORD	ERR0	
4108	024270	005037	002250		CLR	INTFLG	;CLEAR INT. FLAG
4109	024274				CKLOOP		
4110	024274	104406			TRAP	C\$CLP1	
4111	024276	004537	014600		JSR	R5,CHERR	;CHECK CONTROLLER FOR ERRORS
4112	024302				CKLOOP		
4113	024302	104406			TRAP	C\$CLP1	
4114	024304	012737	005756	002306	MOV	#BUF1+1776,GDDAT	
4115	024312	013737	002234	002310	MOV	E.BA,BDDAT	
4116	024320	023737	002306	002310	CMP	GDDAT,BDDAT	;TEST BA REGISTER
4117	024326	001404			BEQ	1\$	
4118	024330				ERRDF	42.,EM10,ERR4	;DATA WRONG IN BA REGISTER
4119	024330	104455			TRAP	C\$ERDF	
4120	024332	000052			.WORD	42	
4121	024334	007046			.WORD	EM10	
4122	024336	012026			.WORD	ERR4	
4123	024340				CKLOOP		;CHECK FOR LOOP MODE
4124	024340	104406			TRAP	C\$CLP1	
4125	024342	013737	002224	002306	MOV	B.DA,GDDAT	;GET BEFORE DA REGISTER
4126	024350	013737	002236	002310	MOV	E.DA,BDDAT	
4127	024356	005037	002274		CLR	TEMP1	
4128	024362	113737	002224	002274	MOVB	B.DA,TEMP1	
4129	024370	062737	000006	002274	ADD	#6,TEMP1	;+6 TO DA LOW BYTE
4130	024376	113737	002274	002306	MOVB	TEMP1,GDDAT	;STORE LOW BYTE OF DA
4131	024404	023737	002306	002310	CMP	GDDAT,BDDAT	
4132	024412	001404			BEQ	2\$	
4133	024414				ERRDF	43.,EM12,ERR4	
4134	024414	104455			TRAP	C\$ERDF	
4135	024416	000053			.WORD	43	
4136	024420	007150			.WORD	EM12	
4137	024422	012026			.WORD	ERR4	
4138	024424				CKLOOP		
4139	024424	104406			TRAP	C\$CLP1	
4140	024426	013737	002314	002306	MOV	GDCRCA,GDDAT	;GET CRC OF DA+3 VALUE
4141	024434	013737	002240	002310	MOV	E.MP,BDDAT	;GET CONTROLLER CRC OF DA+3
4142	024442	023737	002306	002310	CMP	GDDAT,BDDAT	
4143	024450	001404			BEQ	3\$	
4144	024452				ERRDF	44.,EM20,ERR4	
4145	024452	104455			TRAP	C\$ERDF	
4146	024454	000054			.WORD	44	
4147	024456	007400			.WORD	EM20	
4148	024460	012026			.WORD	ERR4	
4149	024462				CKLOOP		
4150	024462	104406			TRAP	C\$CLP1	
4151	024464	013737	002316	002306	MOV	GDCRCB,GDDAT	
4152	024472	013737	002242	002310	MOV	E.MP1,BDDAT	

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 77
 CVRLAC.P'1 31-AUG-82 11:25 **TEST 33** - TEST RLV11 MAINT. FUNCTION -INTERRUPT MODE

SEQ 0077

4153	024500	023737	002306	002310	CMP	GDDAT,BDDAT	
4154	024506	001404			BEQ	4\$	
4155	024510				ERRDF	45,EM21,ERR4	
4156	024510	104455			TRAP	C\$ERDF	
4157	024512	000055			.WORD	45	
4158	024514	007453			.WORD	EM21	
4159	024516	012026			.WORD	ERR4	
4160	024520				4\$: CKLOOP		
4161	024520	104406			TRAP	C\$CLP1	
4162	024522	005037	002360		CLR	SAVCNT	:CLEAR BAD WORD COUNTER
4163	024526	005037	002304		CLR	CHECK	:CLEAR PRINT HEADER INDICATOR
4164	024532	012704	003760		MOV	#BUF1,R4	:GOOD DATA BUFFER
4165	024536	012702	004760		MOV	#BUF2,R2	:DATA BUFFER WRITTEN INTO BY MAINT.
4166	024542	012701	000377		MOV	#255,R1	
4167	024546	011437	002306		5\$: MOV	(R4),GDDAT	:EXPECTED DATA
4168	024552	011237	002310		MOV	(R2),BDDAT	:GET DATA FROM BUFFER
4169	024556	023737	002306	002310	CMP	GDDAT,BDDAT	
4170	024564	001440			BEQ	7\$	:DATA COMPARE
4171	024566	010237	002276		MOV	R2,TMP0	:DATA ERR-GET ADDRESS
4172	024572	005237	002360		INC	SAVCNT	:INC. BAD WORD COUNT
4173	024576	005737	002304		TST	CHECK	:CHECK IF FIRST TIME
4174	024602	001007			BNE	6\$	
4175	024604				ERRDF	46,EM22,ERR3	
4176	024604	104455			TRAP	C\$ERDF	
4177	024606	000056			.WORD	46	
4178	024610	007535			.WORD	EM22	
4179	024612	011724			.WORD	ERR3	
4180	024614	005237	002304		INC	CHECK	:PRINT HEADER ONCE
4181	024620	000422			BR	7\$	
4182	024622				6\$: PRINTX	#FRMT14,E.BA,E.DA,TMP0,GDDAT,BDDAT	
4183	024622	013746	002310		MOV	BDDAT,-(SP)	
4184	024626	013746	002306		MOV	GDDAT,-(SP)	
4185	024632	013746	002276		MOV	TMP0,-(SP)	
4186	024636	013746	002236		MOV	E.DA,-(SP)	
4187	024642	013746	002234		MOV	E.BA,-(SP)	
4188	024646	012746	013477		MOV	#FRMT14,-(SP)	
4189	024652	012746	000006		MOV	#6,-(SP)	
4190	024656	010600			MOV	SP,R0	
4191	024660	104415			TRAP	C\$PNTX	
4192	024662	062706	000016		ADD	#16,SP	
4193	024666				7\$: CKLOOP		
4194	024666	104406			TRAP	C\$CLP1	
4195	024670	005722			TST	(R2)+	:INCREMENT BUFFER
4196	024672	005724			TST	(R4)+	:INCREMENT BUFFER
4197	024674	005301			DEC	R1	:FINISHED BUFFER?
4198	024676	001323			BNE	5\$	:RETURN FOR NEXT COMPARE
4199	024700	005737	002304		TST	CHECK	:CHECK ERROR HEADER FLAG
4200	024704	001412			BEQ	77\$	
4201	024706				PRINTB	#FRMT98,SAVCNT	:PRINT BAD WORD COUNT
4202	024706	013746	002360		MOV	SAVCNT,-(SP)	
4203	024712	012746	013052		MOV	#FRMT98,-(SP)	
4204	024716	012746	000002		MOV	#2,-(SP)	
4205	024722	010600			MOV	SP,R0	
4206	024724	104414			TRAP	C\$PNTB	
4207	024726	062706	000006		ADD	#6,SP	
4208	024732	012737	123456	002306	77\$: MOV	#123456,GDDAT	:EXPECTED DATA IN LAST WORD+1

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 78
 (VRLAC.P11 31-AUG-82 11:25 **TEST 33** - TEST RLV11 MAINT. FUNCTION - INTERRUPT MODE

SEQ 0078

```

4209 024740 011237 002310      MOV      (R2),BDDAT      ;GET LAST WORD+1 FROM BUF2
4210 024744 023737 002306 002310  CMP      GDDAT,BDDAT
4211 024752 001404      BEQ      8$
4212 024754      ERRDF  47,EM23,ERR4
4213 024754 104455      TRAP     C$ERRDF
4214 024756 000057      .WORD    47
4215 024760 007624      .WORD    EM23
4216 024762 012026      .WORD    ERR4
4217 024764      8$:      CKLOOP
4218 024764 104406      TRAP     C$CLP1
4219 024766      ENDSEG
4220 024766      10000$:
4221 024766 104405      TRAP     C$ESEG
4222 024770 005723      TST      (R3)+      ;INC. CRC PATTERN
4223 024772 062737 000002 002352  ADD      #2,PATSAV    ;UPDATE PATTERN TABLE
4224 025000 020327 003064      CMP      R3,#CRCEND  ;CHECK FOR END
4225 025004 001402      BEQ      10$      ;END OF TEST
4226 025006 000137 024162      JMP      101$      ;CONTINUE TEST
4227
4228 025012      10$:
4229
4230 025012      ENDTST
4231 025012      L10063:
4232 025012 104401      TRAP     C$ETST
4233
4234      .SBTTL  **TEST 34** - RLV11 FIFO ADDRESS TEST
4235
4236 025014      BGNTST
4237
4238 025014      STARS
4239      ;*****
4240      ;TEST THAT FIFO OPERATES CORRECTLY.STORE ADDRESS PATTERN
4241      ;IN BUF1 (0-255) THAT CONTAINS A UNIQUE PATTERN IN EACH LOCATION.
4242      ;PERFORM MAINTENANCE FUNCTION AND TEST BUF2 FOR PROPER FIFO
4243      ;ADDRESSING.
4244 025014      STARS
4245      ;*****
4246 025014 005737 002332      TST      T.CNTRL      ;RLV11 OR RLV11
4247 025020 001402      BEQ      1$      ;RLV11;PERFORM TEST
4248 025022 000137 025360      JMP      10$      ;RLV11;SKIP TEST
4249 025026 005001      1$:      CLR      R1
4250 025030 012702 000400      MOV      #256.,R2
4251 025034 012703 003760      MOV      #BUF1,R3      ;SETUP TO STORE PATTERN IN BUF1
4252 025040 010123      2$:      MOV      R1,(R3)+
4253 025042 005201      INC      R1      ;INC. PATTERN
4254 025044 005302      DEC      R2
4255 025046 001374      BNE      2$
4256 025050 012702 000400      MOV      #256.,R2      ;SETUP TO CLEAR BUF2
4257 025054 012703 004760      MOV      #BUF2,R3
4258 025060 005023      3$:      CLR      (R3)+
4259 025062 005302      DEC      R2
4260 025064 001375      BNE      3$
4261 025066 005037 002250      CLR      INTFLG      ;CLEAR INT. FLAG
4262 025072      SETPR1  #PR100
4263 025072 012700 000000      MOV      #PR100,R0
4264 025076 104441      TRAP     C$SPRI

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 79
 (VRLAC.P11 31-AUG-82 11:25 **TEST 34** - RLV11 FIFO ADDRESS TEST

SEQ 0079

4265	025100	004537	016114		JSR	R5,LDFUN	:LOAD FUNCTION
4266	025104	000100			MAINT:INTEN		:MAINT. WITH INTERRUPT
4267	025106	177001			-511.		:WORD COUNT
4268	025110	006364			MATINT		:MAINT. MESSAGE
4269	025112	004537	016724		JSR	R5,WTCRDY	:WAIT FOR READY
4270	025116				CKLOOP		
4271	025116	104406			TRAP	C\$CLP1	
4272	025120				SETPRI	#PRI07	
4273	025120	012700	000340		MOV	#PRI07,R0	
4274	025124	104441			TRAP	C\$SPRI	
4275	025126	005737	002250		TST	INTFLG	:CHECK FOR INTERRUPT
4276	025132	001004			BNE	4\$	
4277	025134				ERRDF	48.,EM24,ERRO	
4278	025134	104455			TRAP	C\$ERDF	
4279	025136	000060			.WORD	48	
4280	025140	007664			.WORD	EM24	
4281	025142	011632			.WORD	ERRO	
4282	025144	005037	002250	4\$:	CLR	INTFLG	
4283	025150				CKLOOP		
4284	025150	104406			TRAP	C\$CLP1	
4285	025152	005037	002360		CLR	SAVCNT	:CLEAR 3AD WORD COUNTER
4286	025156	005037	002304		CLR	CHECK	:CLEAR ERROR HEADER FLAG
4287	025162	005001			CLR	R1	
4288	025164	012702	000377		MOV	#255.,R2	
4289	025170	012703	004760		MOV	#BUF2,R3	
4290	025174	010137	002306	5\$:	MOV	R1,GDDAT	:EXPECTED DATA
4291	025200	011337	002310		MOV	(R3),BDDAT	:DATA IN BUFFER
4292	025204	023737	002306	002310	CMP	GDDAT,BDDAT	
4293	025212	001440			BEQ	7\$	
4294	025214	010337	002276		MOV	R3,TMPO	:GET ADDRESS FOR PRINTOUT
4295	025220	005237	002360		INC	SAVCNT	:INC. BAD WORD COUNTER
4296	025224	005737	002304		TST	CHECK	:CHECK ERROR HEADER FLAG
4297	025230	001007			BNE	6\$	
4298	025232				ERRDF	49.,EM25,ERR3	
4299	025232	104455			TRAP	C\$ERDF	
4300	025234	000061			.WORD	49	
4301	025236	007726			.WORD	EM25	
4302	025240	011724			.WORD	ERR3	
4303	025242	005237	002304		INC	CHECK	
4304	025246	000422			BR	7\$	
4305	025250			6\$:	PRINTX	#FRMT14,E.BA,E.DA,TMPO,GDDAT,BDDAT	
4306	025250	013746	002310		MOV	BDDAT,-(SP)	
4307	025254	013746	002306		MOV	GDDAT,-(SP)	
4308	025260	013746	002276		MOV	TMPO,-(SP)	
4309	025264	013746	002236		MOV	E.DA,-(SP)	
4310	025270	013746	002234		MOV	E.BA,-(SP)	
4311	025274	012746	013477		MOV	#FRMT14,-(SP)	
4312	025300	012746	000006		MOV	#6,-(SP)	
4313	025304	010600			MOV	SP,R0	
4314	025306	104415			TRAP	C\$PNTX	
4315	025310	062706	000016		ADD	#16,SP	
4316	025314			7\$:	CKLOOP		
4317	025314	104406			TRAP	C\$CLP1	
4318	025316	005723			TST	(R3)+	
4319	025320	005201			INC	R1	:UPDATE PATTERN EXPECTED
4320	025322	005302			DEC	R2	

4321	025324	001323		BNE	5\$	
4322	025326	005737	002304	TST	CHECK	;CHECK ERROR FLAG
4323	025332	001412		BEQ	10\$	
4324	025334			PRINTB	#FRMT98,SAVCNT	;PRINT NUMBER OF BAD WORDS
4325	025334	013746	002360	MOV	SAVCNT,-(SP)	
4326	025340	012746	013052	MOV	#FRMT98,-(SP)	
4327	025344	012746	000002	MOV	#2,-(SP)	
4328	025350	010600		MOV	SP,RO	
4329	025352	104414		TRAP	C\$PNTB	
4330	025354	062706	000006	ADD	#6,SP	
4331						
4332	025360			10\$:		
4333						
4334	025360			ENDTST		
4335	025360			L10064:		
4336	025360	104401		TRAP	C\$ETST	
4337						
4338				.SBTTL	**TEST 35** - RLV11 FIFO ADDRESS COMPLEMENT TEST	
4339						
4340	025362			BGNTST		
4341						
4342	025362			STARS		
4343				;	*****	
4344				;	TEST THAT FIFO OPERATES CORRECTLY. STORE ADDRESS COMPLEMENT PAT.	
4345				;	IN BUF1 (0-255) THAT CONTAINS A UNIQUE PATTERN IN EACH LOCATION.	
4346				;	PERFORM MAINTENANCE FUNCTION AND TEST BUF2 FOR PROPER FIFO	
4347				;	ADDRESSING.	
4348	025362			STARS		
4349				;	*****	
4350	025362	005737	002332	TST	T.CNTRL	;RL11 OR RLV11
4351	025366	001402		BEQ	1\$	;RLV11;PERFORM TEST
4352	025370	000137	025732	JMP	10\$	;RL11;SKIP TEST
4353	025374	012701	177777	1\$:	MOV	#177777,R1
4354	025400	012702	000400		MOV	#256.,R2
4355	025404	012703	003760		MOV	#BUF1,R3
4356	025410	010123		2\$:	MOV	R1,(R3)+
4357	025412	005301			DEC	R1
4358	025414	005302			DEC	R2
4359	025416	001374			BNE	2\$
4360	025420	012702	000400		MOV	#256.,R2
4361	025424	012703	004760		MOV	#BUF2,R3
4362	025430	005023		3\$:	CLR	(R3)+
4363	025432	005302			DEC	R2
4364	025434	001375			BNE	3\$
4365	025436	005037	002250		CLR	INTFLG
4366	025442				SETPRI	#PRI00
4367	025442	012700	000000		MOV	#PRI00,RO
4368	025446	104441			TRAP	C\$SPRI
4369	025450	004537	016114		JSR	R5,LDFUN
4370	025454	000100			MAINT!INTEN	;LOAD FUNCTION
4371	025456	177001			-511.	;MAINT. WITH INTERRUPT
4372	025460	006364			MATINT	;WORD COUNT
4373	025462	004537	016724		JSR	R5,WTCRDY
4374	025466				CKLOOP	;MAINT. MESSAGE
4375	025466	104406			TRAP	C\$CLP1
4376	025470				SETPRI	#PRI07

4377	025470	012700	000340	MOV	#PRI07,R0	
4378	025474	104441		TRAP	C\$SPRI	
4379	025476	005737	002250	TST	INTFLG	;CHECK FOR INTERRUPT
4380	025502	001004		BNE	4\$	
4381	025504			ERRDF	50.,EM24,ERR0	
4382	025504	104455		TRAP	C\$ERDF	
4383	025506	000062		.WORD	50	
4384	025510	007664		.WORD	EM24	
4385	025512	011632		.WORD	ERR0	
4386	025514	005037	002250	4\$: CLR	INTFLG	
4387	025520			CKLOOP		
4388	025520	104406		TRAP	C\$CLP1	
4389	025522	005037	002360	CLR	SAVCNT	;CLEAR BAD WORD COUNTER
4390	025526	005037	002304	CLR	CHECK	;CLEAR ERROR HEADER FLAG
4391	025532	012701	177777	MOV	#177777,R1	
4392	025536	012702	000377	MOV	#255.,R2	
4393	025542	012703	004760	MOV	#BUF2,R3	
4394	025546	010137	002306	5\$: MOV	R1,GDDAT	;EXPECTED DATA
4395	025552	011337	002310	MOV	(R3),BDDAT	;DATA IN BUFFER
4396	025556	023737	002306	002310	CMP	GDDAT,BDDAT
4397	025564	001440		BEQ	7\$	
4398	025566	010337	002276	MOV	R3,TMPO	;GET ADDRESS FOR PRINTOUT
4399	025572	005237	002360	INC	SAVCNT	;INC. BAD WORD COUNTER
4400	025576	005737	002304	TST	CHECK	;CHECK ERROR HEADER FLAG
4401	025602	001007		BNE	6\$	
4402	025604			ERRDF	51.,EM26,ERR3	
4403	025604	104455		TRAP	C\$ERDF	
4404	025606	000063		.WORD	51	
4405	025610	007765		.WORD	EM26	
4406	025612	011724		.WORD	ERR3	
4407	025614	005237	002304	INC	CHECK	
4408	025620	000422		BR	7\$	
4409	025622			6\$: PRINTX	#FRMT14,E.BA,E.DA,TMPO,GDDAT,BDDAT	
4410	025622	013746	002310	MOV	BDDAT,-(SP)	
4411	025626	013746	002306	MOV	GDDAT,-(SP)	
4412	025632	013746	002276	MOV	TMPO,-(SP)	
4413	025636	013746	002236	MOV	E.DA,-(SP)	
4414	025642	013746	002234	MOV	E.BA,-(SP)	
4415	025646	012746	013477	MOV	#FRMT14,-(SP)	
4416	025652	012746	000006	MOV	#6,-(SP)	
4417	025655	010600		MOV	SP,R0	
4418	025660	104415		TRAP	C\$PNTX	
4419	025662	062706	000016	ADD	#16,SP	
4420	025666			7\$: CKLOOP		
4421	025666	104406		TRAP	C\$CLP1	
4422	025670	005723		TST	(R3)+	
4423	025672	005301		DEC	R1	;GET NEXT PATTERN
4424	025674	005302		DEC	R2	
4425	025676	001323		BNE	5\$	
4426	025700	005737	002304	TST	CHECK	;CHECK ERROR FLAG
4427	025704	001412		BEQ	10\$	
4428	025706			PRINTB	#FRMT98,SAVCNT	;PRINT NO. OF BAD WORDS
4429	025706	013746	002360	MOV	SAVCNT,-(SP)	
4430	025712	012746	013052	MOV	#FRMT98,-(SP)	
4431	025716	012746	000002	MOV	#2,-(SP)	
4432	025722	010600		MOV	SP,R0	

.MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 82
 CVRLAC.P11 31-AUG-82 11:25 **TEST 35** - RLV11 FIFO ADDRESS COMPLEMENT TEST

SEQ 0082

```

4433 025724 104414          TRAP    C$PNTB
4434 025726 062706 000006    ADD     #6.SP
4435
4436 025732          10$:
4437 025732          ENDTST
4438 025732          L10065:
4439 025732 104401          TRAP    C$ETST
4440
4441
4442          .SBTTL  **TEST 36** - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE
4443
4444 025734          BGNSTST          ;****START OF TEST****
4445
4446 025734          STARS
4447          ;*****
4448          ;PERFORM RLV11 MAINT. FUNCTION WITH COMPLEMENT DATA PATTERNS IN BUF1
4449          ;CHECK FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
4450          ;WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER
4451          ;THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
4452          ;RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
4453          ;FIFO INTO BUF2 MEMORY FOR PROPER DATA.
4454          ;CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
4455          ;VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.
4456 025734          STARS
4457          ;*****
4458 025734 005737 002332          TSI      T.CNTRL          ;RLV11?
4459 025740 001402          BEQ      100$          ;YES,RLV11
4460 025742 000137 026610          JMP      10$          ;NO,SKIP TEST
4461 025746 012703 002774          100$: MOV     #PATCRC,R3          ;GET CRC PATTERN
4462 025752 012737 003066 002352  MOV     #PATDAT,PATSAV ;GET DATA PATTERN
4463 025760 011337 025776          101$: MOV     (R3),102$
4464 025764 017737 154362 026006  MOV     @PATSAV,103$
4465 025772 004537 015514          JSR      R5,CALCRC          ;CALCULATE CRC BEFORE TEST
4466 025776 000000          102$: .WORD    0          ;PATTERN FOR CRC TEST
4467 026000
4468 026000 104404          BGNSEG
4469 026002 001537 016214          TRAP    C$BSEG
4470 026006 000000          JSR      R5,SETCMP          ;SETUP PATTERN IN BUFFER
4471 026010          .WORD    0          ;BUFFER PATTERN
4472 026010 012700 000000          SETPRI  #PRI00          ;SET PRIORITY TO ZERO
4473 026014 104441          MOV     #PRI00,R0
4474 026016 005037 002250          TRAP    C$SPRI
4475 026022 004537 016114          CLR     INTFLG          ;CLEAR INT. FLAG
4476 026026 000100          JSR      R5,LDFUN          ;PERFORM MAINT. FUNCTION
4477 026030 177001          MAINT!INTEN ;MAINT FUNCTION INT. DRIVEN
4478 026032 006364          -511.          ;WORD COUNT
4479 026034 004537 016724          MATINT ;MESSAGE
4480 026040          JSR      R5,WTCRDY          ;WAIT FOR READY
4481 026040 104406          CKLOOP
4482 026042          TRAP    C$CLP1
4483 026042 012700 000340          SETPRI  #PRI07
4484 026046 104441          MOV     #PRI07,R0
4485 026050 005737 002250          TRAP    C$SPRI
4486 026054 001004          TST     INTFLG
4487 026056          BNE     104$
4488 026056 104455          ERRDI  52.,EM24,ERRO
          TRAP    C$ERDF

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 83
 CVRLAC.P11 31-AUG-82 11:25 **TEST 36** - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE

SEQ 0083

4489	026060	000064			.WORD	52	
4490	026062	007664			.WORD	EM24	
4491	026064	011632			.WORD	ERR0	
4492	026066	005037	002250		104\$: CLR	INTFLG	:CLEAR INT. FLAG
4493	026072				CKLOOP		
4494	026072	104406			TRAP	C\$CLP1	
4495	026074	004537	014600		JSR	R5,CHERR	:CHECK CONTROLLER FOR ERRORS
4496	026100				CKLOOP		
4497	026100	104406			TRAP	C\$CLP1	
4498	026102	012737	005756	002306	MOV	#SUF1+1776,GDDAT	
4499	026110	013737	002234	002310	MOV	E.BA,BDDAT	
4500	026116	023737	002306	002310	CMP	GDDAT,BDDAT	:TEST BA REGISTER
4501	026124	001404			BEQ	1\$	
4502	026126				ERRDF	53.,EM10,ERR4	:DATA WRONG IN BA REGISTER
4503	026126	104455			TRAP	C\$ERDF	
4504	026130	000065			.WORD	53	
4505	026132	007046			.WORD	EM10	
4506	026134	012026			.WORD	ERR4	
4507	026136				1\$: CKLOOP		:CHECK FOR LOOP MODE
4508	026136	104406			TRAP	C\$CLP1	
4509	026140	013737	002224	002306	MOV	B.DA,GDDAT	:GET BEFORE DA REGISTER
4510	026146	013737	002236	002310	MOV	E.DA,BDDAT	
4511	026154	005037	002274		CLR	TEMP1	
4512	026160	113737	002224	002274	MOVB	B.DA,TEMP1	
4513	026166	062737	000006	002274	ADD	#6,TEMP1	:+6 TO DA LOW BYTE
4514	026174	113737	002274	002306	MOVB	TEMP1,GDDAT	:STORE LOW BYTE OF DA
4515	026202	023737	002306	002310	CMP	GDDAT,BDDAT	
4516	026210	001404			BEQ	2\$	
4517	026212				ERRDF	54.,EM12,ERR4	
4518	026212	104455			TRAP	C\$ERDF	
4519	026214	000066			.WORD	54	
4520	026216	007150			.WORD	EM12	
4521	026220	012026			.WORD	ERR4	
4522	026222				2\$: CKLOOP		
4523	026222	104406			TRAP	C\$CLP1	
4524	026224	013737	002314	002306	MOV	GDCRCA,GDDAT	:GET CRC OF DA+3 VALUE
4525	026232	013737	002240	002310	MOV	E.MP,BDDAT	:GET CONTROLLER CRC OF DA+3
4526	026240	023737	002306	002310	CMP	GDDAT,BDDAT	
4527	026246	001404			BEQ	3\$	
4528	026250				ERRDF	55.,EM20,ERR4	
4529	026250	104455			TRAP	C\$ERDF	
4530	026252	000067			.WORD	55	
4531	026254	007400			.WORD	EM20	
4532	026256	012026			.WORD	ERR4	
4533	026260				3\$: CKLOOP		
4534	026260	104406			TRAP	C\$CLP1	
4535	026262	013737	002316	002306	MOV	GDCRCB,GDDAT	
4536	026270	013737	002242	002310	MOV	E.MP1,BDDAT	
4537	026276	023737	002306	002310	CMP	GDDAT,BDDAT	
4538	026304	001404			BEQ	4\$	
4539	026306				ERRDF	56.,EM21,ERR4	
4540	026306	104455			TRAP	C\$ERDF	
4541	026310	000070			.WORD	56	
4542	026312	007453			.WORD	EM21	
4543	026314	012026			.WORD	ERR4	
4544	026316				4\$: CKLOOP		

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 84
 CVRLAC.P11 31-AUG-82 11:25 **TEST 36** - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE

SEQ 0084

4545	026316	104406		TRAP	C\$CLP1	
4546	026320	005037	002360	CLR	SAVCNT	;CLEAR BAD WORD COUNTER
4547	026324	005037	002304	CLR	CHECK	;CLEAR PRINT HEADER INDICATOR
4548	026330	012704	003760	MOV	#BUF1,R4	;GOOD DATA BUFFER
4549	026334	012702	004760	MOV	#BUF2,R2	;DATA BUFFER WRITTEN INTO BY MAINT.
4550	026340	012701	000377	MOV	#255,R1	
4551	026344	011437	002306	5\$: MOV	(R4),GDDAT	;EXPECTED DATA
4552	026350	011237	002310	MOV	(R2),BDDAT	;GET DATA FROM BUFFER
4553	026354	023737	002306	002310 CMP	GDDAT,BDDAT	
4554	026362	001440		BEQ	7\$	;DATA COMPARE
4555	026364	010237	002276	MOV	R2,TMP0	;DATA ERR-GET ADDRESS
4556	026370	005237	002360	INC	SAVCNT	;INC. BAD WORD COUNTER
4557	026374	005737	002304	TST	CHECK	;CHECK IF FIRST TIME
4558	026400	001007		BNE	6\$	
4559	026402			ERRDF	57,EM22,ERR3	
4560	026402	104455		TRAP	C\$ERDF	
4561	026404	000071		.WORD	57	
4562	026406	007535		.WORD	EM22	
4563	026410	011724		.WORD	ERR3	
4564	026412	005237	002304	INC	CHECK	;PRINT HEADER ONCE
4565	026416	000422		BR	7\$	
4566	026420			6\$: PRINTX	#FRMT14,E.BA,E.DA,TMP0,GDDAT,BDDAT	
4567	026420	013746	002310	MOV	BDDAT,-(SP)	
4568	026424	013746	002306	MOV	GDDAT,-(SP)	
4569	026430	013746	002276	MOV	TMP0,-(SP)	
4570	026434	013746	002236	MOV	E.DA,-(SP)	
4571	026440	013746	002234	MOV	E.BA,-(SP)	
4572	026444	012746	013477	MOV	#FRMT14,-(SP)	
4573	026450	012746	000006	MOV	#6,-(SP)	
4574	026454	010600		MOV	SP,R0	
4575	026456	104415		TRAP	C\$PNTX	
4576	026460	062706	000016	ADD	#16,SP	
4577	026464			7\$: CKLOOP		
4578	026464	104406		TRAP	C\$CLP1	
4579	026466	005722		TST	(R2)+	;INCREMENT BUFFER
4580	026470	005724		TST	(R4)+	;INCREMENT BUFFER
4581	026472	005301		DEC	R1	;FINISHED BUFFER?
4582	026474	001323		BNE	5\$	;RETURN FOR NEXT COMPARE
4583	026476	005737	002304	TST	CHECK	;CHECK ERROR FLAG
4584	026502	001412		BEQ	77\$	
4585	026504			PRINTB	#FRMT98,SAVCNT	;PRINT NO OF BAD WORDS
4586	026504	013746	002360	MOV	SAVCNT,-(SP)	
4587	026510	012746	013052	MOV	#FRMT98,-(SP)	
4588	026514	012746	000002	MOV	#2,-(SP)	
4589	026520	010600		MOV	SP,R0	
4590	026522	104414		TRAP	C\$PNTB	
4591	026524	062706	000006	ADD	#6,SP	
4592	026530	012737	123456	002306 77\$: MOV	#123456,GDDAT	;EXPECTED DATA IN LAST WORD+1
4593	026536	011237	002310	MOV	(R2),BDDAT	;GET LAST WORD+1 FROM BUF2
4594	026542	023737	002306	002310 CMP	GDDAT,BDDAT	
4595	026550	001404		BEQ	8\$	
4596	026552			ERRDF	58,EM23,ERR4	
4597	026552	104455		TRAP	C\$ERDF	
4598	026554	000072		.WORD	58	
4599	026556	007624		.WORD	EM23	
4600	026560	012026		.WORD	ERR4	

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 85
 CVRLAC.P11 31-AUG-82 11:25

TEST 36 - TEST RLV11 MAINT. WITH COMPLEMENT DATA -INT. MODE

SEQ 0085

```

4601 026562      RS:      CKLOOP
4602 026562      TRAP      C$CLP1
4603 026564      ENDSEG
4604 026564      10000$:
4605 026564      TRAP      C$ESEG
4606 026566      TST      (R3)+      ;INC. CRC PATTERN
4607 026570      062737      000002      002352      ADD      #2,PATSAV      ;UPDATE PATTERN TABLE
4608 026576      C20327      003064      CMP      R3,#CRCEND      ;CHECK FOR END
4609 026602      001402      BEQ      10$      ;END OF TEST
4610 026604      000137      025760      JMP      101$      ;CONTINUE TEST
4611
4612 026610      10$:
4613 026610      ENDIST
4614 026610      L10066:
4615 026610      104401      TRAP      C$ETST
4616
4617      .SBTTL      **TEST 37** - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE
4618
4619 026612      BGNSTST      ;****START OF TEST****
4620
4621 026612      STARS
4622      ;*****
4623      ;PERFORM RLV11 MAINT. FUNCTION WITH RANDOM DATA PATTERNS IN BUF1
4624      ;RANDOM PATTERN IS THE SAME FOR EACH CONTROLLER UNDER TEST.
4625      ;RANDOM PATTERN WILL CHANGE AT END OF PASS.
4626      ;RANDOM PATTERN WILL INIT AT START OR RESTART.
4627      ;CHECK FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
4628      ;WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER
4629      ;THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
4630      ;RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
4631      ;FIFO INTO BUF2 MEMORY FOR PROPER DATA.
4632      ;CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
4633      ;VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.
4634 026612      STARS
4635      ;*****
4636 026612      005737      002332      TST      T.CNTRL      ;RLV11?
4637 026616      001402      BEQ      101$      ;YES,RLV11
4638 026620      000137      027432      JMP      10$      ;NO,SKIP TEST
4639 026624      013737      002346      026644      101$:      MOV      TEMLO,102$      ;STARTING RANDOM PATTERN
4640 026632      013737      002346      002344      MOV      TEMLO,LONUM      ;RESET RANDOM START
4641 026640      004537      015514      JSR      R5,CALCRC      ;CALCULATE CRC BEFORE TEST
4642 026644      000000      102$:      .WORD      0      ;PATTERN FOR CRC TEST
4643 026646      004537      016306      JSR      R5,SETRAN      ;SETUP RANDOM PATTERN IN BUFFER
4644 026652      BGNSEG
4645 026652      104404      TRAP      C$BSEG
4646 026654      SETPRI      #PRI00      ;SET PRIORITY TO ZERO
4647 026654      012700      000000      MOV      #PRI00,R0
4648 026660      104441      TRAP      C$SPRI
4649 026662      005037      002250      CLR      INTFLG      ;CLEAR INT. FLAG
4650 026666      004537      016114      JSR      R5,LDFUN      ;PERFORM MAINT. FUNCTION
4651 026672      000100      MAINT!INTEN      ;MAINT FUNCTION INT. DRIVEN
4652 026674      177001      -511.      ;WORD COUNT
4653 026676      006364      MATINT      ;MESSAGE
4654 026700      004537      016724      JSR      R5,WTCRDY      ;WAIT FOR READY
4655 026704      CKLOOP
4656 026704      104406      TRAP      C$CLP1

```

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 86
 CVRLAC.P11 31-AUG-82 11:25 **TEST 37** - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

SEQ 0086

4657	026706				SETPRI	#PRI07	
4658	026706	012700	000340		MOV	#PRI07,R0	
4659	026712	104441			TRAP	C\$SPRI	
4660	026714	005737	002250		TST	INTFLG	
4661	026720	001004			BNE	104\$	
4662	026722				ERRDF	59.,EM24,ERR0	
4663	026722	104455			TRAP	C\$ERDF	
4664	026724	000073			.WORD	59	
4665	026726	007664			.WORD	EM24	
4666	026730	011632			.WORD	ERR0	
4667	026732	005037	002250		104\$: CLR	INTFLG	;CLEAR INT. FLAG
4668	026736				CKLOOP		
4669	026736	104406			TRAP	C\$CLP1	
4670	026740	004537	014600		JSR	R5,CHERR	;CHECK CONTROLLER FOR ERRORS
4671	026744				CKLOOP		
4672	026744	104406			TRAP	C\$CLP1	
4673	026746	012737	005756	002306	MOV	#BUF1+1776,GDDAT	
4674	026754	013737	002234	002310	MOV	E.BA,BDDAT	
4675	026762	023737	002306	002310	CMP	GDDAT,BDDAT	;TEST BA REGISTER
4676	026770	001404			BEQ	1\$	
4677	026772				ERRDF	60.,EM10,ERR4	;DATA WRONG IN BA REGISTER
4678	026772	104455			TRAP	C\$ERDF	
4679	026774	000074			.WORD	60	
4680	026776	007046			.WORD	EM10	
4681	027000	012026			.WORD	ERR4	
4682	027002				1\$: CKLOOP		;CHECK FOR LOOP MODE
4683	027002	104406			TRAP	C\$CLP1	
4684	027004	013737	002224	002306	MOV	B.DA,GDDAT	;GET BEFORE DA REGISTER
4685	027012	013737	002236	002310	MOV	E.DA,BDDAT	
4686	027020	005037	002274		CLR	TEMP1	
4687	027024	113737	002224	002274	MOVB	B.DA,TEMP1	
4688	027032	062737	000006	002274	ADD	#6,TEMP1	;+6 TO DA LOW BYTE
4689	027040	113737	002274	002306	MOVB	TEMP1,GDDAT	;STORE LOW BYTE OF DA
4690	027046	023737	002306	002310	CMP	GDDAT,BDDAT	
4691	027054	001404			BEQ	2\$	
4692	027056				ERRDF	61.,EM12,ERR4	
4693	027056	104455			TRAP	C\$ERDF	
4694	027060	000075			.WORD	61	
4695	027062	007150			.WORD	EM12	
4696	027064	012026			.WORD	ERR4	
4697	027066				2\$: CKLOOP		
4698	027066	104406			TRAP	C\$CLP1	
4699	027070	013737	002314	002306	MOV	GDCRCA,GDDAT	;GET CRC OF DA+3 VALUE
4700	027076	013737	002240	002310	MOV	E.MP,BDDAT	;GET CONTROLLER CRC OF DA+3
4701	027104	023737	002306	002310	CMP	GDDAT,BDDAT	
4702	027112	001404			BEQ	3\$	
4703	027114				ERRDF	62.,EM20,ERR4	
4704	027114	104455			TRAP	C\$ERDF	
4705	027116	000076			.WORD	62	
4706	027120	007400			.WORD	EM20	
4707	027122	012026			.WORD	ERR4	
4708	027124				3\$: CKLOOP		
4709	027124	104406			TRAP	C\$CLP1	
4710	027126	013737	002316	002306	MOV	GDCRCB,GDDAT	
4711	027134	013737	002242	002310	MOV	E.MP1,BDDAT	
4712	027142	023737	002306	002310	CMP	GDDAT,BDDAT	

4713	027150	001404		BEQ	4\$	
4714	027152			ERRDF	63.,EM21,ERR4	
4715	027152	104455		TRAP	C\$ERDF	
4716	027154	000077		.WORD	63	
4717	027156	007453		.WORD	EM21	
4718	027160	012026		.WORD	ERR4	
4719	027162		4\$:	CKLOOP		
4720	027162	104406		TRAP	C\$CLP1	
4721	027164	005037	002360	CLR	SAVCNT	:CLEAR BAD WORD COUNTER
4722	027170	005037	002304	CLR	CHECK	:CLEAR PRINT HEADER INDICATOR
4723	027174	012703	003760	MOV	#BUF1,R3	:BUFFER WITH RANDOM NUMBERS
4724	027200	012702	004760	MOV	#BUF2,R2	:DATA BUFFER WRITTEN INTO BY MAINT.
4725	027204	012701	000377	MOV	#255.,R1	
4726	027210	011337	002306	MOV	(R3),GDDAT	:EXPECTED DATA
4727	027214	011237	002310	MOV	(R2),BDDAT	:GET DATA FROM BUFFER
4728	027220	023737	002306	CMP	GDDAT,BDDAT	
4729	027226	001440		BEQ	7\$	:DATA COMPARE
4730	027230	010237	002276	MOV	R2,TMP0	:DATA ERR-GET ADDRESS
4731	027234	005237	002360	INC	SAVCNT	:INC BAD WORD COUNT
4732	027240	005737	002304	TST	CHECK	:CHECK IF FIRST TIME
4733	027244	001007		BNE	6\$	
4734	027246			ERRDF	64.,EM22,ERR3	
4735	027246	104455		TRAP	C\$ERDF	
4736	027250	000100		.WORD	64	
4737	027252	007535		.WORD	EM22	
4738	027254	011724		.WORD	ERR3	
4739	027256	005237	002304	INC	CHECK	:PRINT HEADER ONCE
4740	027262	000422		BR	7\$	
4741	027264		6\$:	PRINTX	#FRMT14,E.BA,E.DA,TMP0,GDDAT,BDDAT	
4742	027264	013746	002310	MOV	BDDAT,-(SP)	
4743	027270	013746	002306	MOV	GDDAT,-(SP)	
4744	027274	013746	002276	MOV	TMP0,-(SP)	
4745	027300	013746	002236	MOV	E.DA,-(SP)	
4746	027304	013746	002234	MOV	E.BA,-(SP)	
4747	027310	012746	013477	MOV	#FRMT14,-(SP)	
4748	027314	012746	000006	MOV	#6,-(SP)	
4749	027320	010600		MOV	SP,R0	
4750	027322	104415		TRAP	C\$PNTX	
4751	027324	062706	000016	ADD	#16,SP	
4752	027330		7\$:	CKLOOP		
4753	027330	104406		TRAP	C\$CLP1	
4754	027332	005722		TST	(R2)+	:INCREMENT BUFFER
4755	027334	005723		TST	(R3)+	:INCREMENT GOOD BUFFER
4756	027336	005301		DEC	R1	:FINISHED BUFFER?
4757	027340	001323		BNE	5\$	:RETURN FOR NEXT COMPARE
4758	027342	005737	002304	TST	CHECK	:CHECK ERROR FLAG
4759	027346	001412		BEQ	77\$	
4760	027350			PRINTB	#FRMT98,SAVCNT	:PRINT NO. OF BAD WORDS
4761	027350	013746	002360	MOV	SAVCNT,-(SP)	
4762	027354	012746	013052	MOV	#FRMT98,-(SP)	
4763	027360	012746	000002	MOV	#2,-(SP)	
4764	027364	010600		MOV	SP,R0	
4765	027366	104414		TRAP	C\$PNTB	
4766	027370	062706	000006	ADD	#6,SP	
4767	027374	012737	123456	MOV	#123456,GDDAT	:EXPECTED DATA IN LAST WORD+1
4768	027402	011237	002310	MOV	(R2),BDDAT	:GET LAST WORD+1 FROM BUF2

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 88
 CVRLAC.P11 31-AUG-82 11:25 **TEST 37** - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

SEQ 0088

4769	027406	023737	002306	002310	CMP	GDDAT,BDDAT
4770	027414	001404			BEQ	8\$
4771	027416				ERRDF	65.,EM23,ERR4
4772	027416	104455			TRAP	C\$ERDF
4773	027420	000101			.WORD	65
4774	027422	007624			.WORD	EM23
4775	027424	012026			.WORD	ERR4
4776	027426		8\$:		CKLOOP	
4777	027426	104406			TRAP	C\$CLP1
4778	027430				ENDSEG	
4779	027430		10000\$:			
4780	027430	104405			TRAP	C\$ESEG
4781						
4782	027432		10\$:			
4783	027432		ENDTST			
4784	027432		L10067:			
4785	027432	104401			TRAP	C\$ETST
4786						
4787	027434		BGNMOD		HRDPRM	
4788						
4789	027434				BGNHRD	
4790	027434	000025			.WORD	L10070-L\$HARD/2
4791						
4792	027436				GPRML	L1YPMS,L1YPE,1,YES
4793	027436	005130			.WORD	T\$CODE
4794	027440	027552			.WORD	L1YPMS
4795	027442	000001			.WORD	1
4796	027444				GPRMA	CSRMSG,CSR,0,160000,177776,YES
4797	027444	000031			.WORD	T\$CODE
4798	027446	027510			.WORD	CSRMSG
4799	027450	160000			.WORD	T\$LOLIM
4800	027452	177776			.WORD	T\$HILIM
4801	027454				GPRMA	VECMSG,VECT,0,0,776,YES
4802	027454	001031			.WORD	T\$CODE
4803	027456	027535			.WORD	VECMSG
4804	027460	000000			.WORD	T\$LOLIM
4805	027462	000776			.WORD	T\$HILIM
4806	027464				GPRMD	DRMSG,DRBT,0,03400,0,7,YES
4807	027464	003032			.WORD	T\$CODE
4808	027466	027544			.WORD	DRMSG
4809	027470	003400			.WORD	03400
4810	027472	000000			.WORD	T\$LOLIM
4811	027474	000007			.WORD	T\$HILIM
4812	027476				GPRMD	BRMSG,PRIOR,0,340,0,7,YES
4813	027476	002032			.WORD	T\$CODE
4814	027500	027524			.WORD	BRMSG
4815	027502	000340			.WORD	340
4816	027504	000000			.WORD	T\$LOLIM
4817	027506	000007			.WORD	T\$HILIM
4818						
4819	027510				ENDHRD	
4820					.EVEN	
4821	027510		L10070:			
4822						
4823	027510	052502	020123	042101	CSRMSG:	.ASCIIZ /BUS ADDRESS/
4824	027516	051104	051505	000123		

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 89
 CVRLAC.P11 31-AUG-82 11:25 **TEST 37** - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

SEQ 0089

```

4825 027524 051102 046040 053105 BRMSG: .ASCIZ /BR LEVEL/
4826 027532 046105 000
4827 027535 126 041505 047524 VECMSG: .ASCIZ /VECTOR/
4828 027542 000122
4829 027544 051104 053111 000105 DRMSG: .ASCIZ /DRIVE/
4830 027552 030461 031057 020063 LTYPMS: .ASCIZ \11/23 PROCESSOR\
4831 027560 051120 041517 051505
4832 027566 047523 000122
4833 .EVEN
4834
4835 027572 ENDMOD
4836
4837
4838
4839 027572 BGNMOD SFTPRM
4840
4841 027572 BGNSFT
4842 027572 000014 .WORD L10071-L$SOFT/2
4843 027574 GPRML DMSG,DLT,1,YES
4844 027574 000130 .WORD T$CODE
4845 027576 027624 .WORD DMSG
4846 027600 000001 .WORD 1
4847 027602 XFERF 1$
4848 027602 006044 .WORD T$CODE
4849 027604 GPRMD EMSG,ELT,0,177777,0,177777,YES
4850 027604 001032 .WORD T$CODE
4851 027606 027661 .WORD EMSG
4852 027610 177777 .WORD 177777
4853 027612 000000 .WORD T$LOLIM
4854 027614 177777 .WORD T$HILIM
4855 027616 1$ GPRML SMSG,SIZE,1,YES
4856 027616 002130 .WORD T$CODE
4857 027620 027650 .WORD SMSG
4858 027622 000001 .WORD 1
4859 027624 ENDSFT
4860 .EVEN
4861 027624 L10071:
4862
4863
4864 027624 051104 050117 047440 DMSG: .ASCIZ /DROP ON ERROR LIMIT/
4865 027650 052501 047524 044523 SMSG: .ASCIZ /AUTOSIZE/
4866 027661 105 051122 051117 EMSG: .ASCIZ /ERROR LIMIT/
4867
4868 027676 .EVEN
4869
4870 027676 ENDMOD
4871
4872 030514 .=30514
4873 030514 000000 :AREA RESERVED AS PATCH AREA FOR DIAGNOSTIC
4874 030516 000000 LASTAD
4875 030520 .EVEN
          .WORD 0
          .WORD 0
L$LAST::

```

.MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 90
CVRLAC.P11 31-AUG-82 11:25 **TEST 37** - TEST RLV11 MAINT. WITH RANDOM DATA -INT. MODE

SEQ 009C

4876
4877

000001

.END

ADDCOD	014510	G	1844#																	
ADR	= 000020	G	1034#																	
AFREG	006501		1342#	1542																
AFTER	015712		2126#	2383	2393															
ARLBA	006436		1342#	1518	1539															
ARLCS	006431		1342#	1520	1541															
ARLDA	006444		1342#	1531	1553															
ARLMP	006452		1342#	1529	1551															
ASSEMB	= 000010		898																	
BATEST	020320		2925#	2946																
BA16	= 000020		1053#																	
BA17	= 000040		1052#																	
BCCFBK	002260		1124#	2303*	2309*	2312*	2315	2319												
BCSR	002210		1104#	1723*	1734															
BDDAT	002310		1136#	1370	1393	1415	1459	1491	2770*	2771*	2772	2806*	2838*	2881*	2882*					
			2883	2931*	2932	2976*	2977	3026*	3027*	3028	3077*	3078*	3079	3126*	3127					
			3171*	3172	3215*	3216	3260*	3261	3315*	3316*	3317	3351*	3383*	3424*	3438*					
			3477*	3478*	3479	3495*	3534*	3535*	3536	3551*	3591*	3592*	3593	3608*	3620*					
			3828*	3830*	3850*	3867	3869	3938*	3939	3949*	3954	3964*	3965	3975*	3976					
			3991*	3992	4006	4032*	4033	4115*	4116	4126*	4131	4141*	4142	4152*	4153					
			4168*	4169	4183	4209*	4210	4291*	4292	4306	4395*	4396	4410	4499*	4500					
			4510*	4515	4525*	4526	4536*	4537	4552*	4553	4567	4593*	4594	4674*	4675					
			4685*	4690	4700*	4701	4711*	4712	4727*	4728	4742	4768*	4769							
BEFORE	015660		2009	2116#	2199															
BEGPAT	002564		1164#	2922	2969	3116	3163	3208	3252											
BEREG	006400		1342#	1521																
BIT0	= 000001	G	1007#	1046	1069	2929														
BIT00	= 000001	G	996#	1007																
BIT01	= 000002	G	995#	1006																
BIT02	= 000004	G	994#	1005																
BIT03	= 000010	G	993#	1004																
BIT04	= 000020	G	992#	1003																
BIT05	= 000040	G	991#	1002																
BIT06	= 000100	G	990#	1001																
BIT07	= 000200	G	989#	1000																
BIT08	= 000400	G	988#	999																
BIT09	= 001000	G	987#	998																
BIT11	= 000002	G	1006#	1060	1062	1064	1066	1068	1301											
BIT10	= 002000	G	986#	1050	2047															
BIT11	= 004000	G	985#	1940	1949															
BIT12	= 010000	G	984#	1944	1953	2042														
BIT13	= 020000	G	983#	1054																
BIT14	= 040000	G	982#	1049																
BIT15	= 100000	G	981#	1048	2037															
BIT2	= 000004	G	1005#	1061	1062	1065	1070	1302												
BIT3	= 000010	G	1004#	1063	1064	1065	1067	1303												
BIT4	= 000020	G	1003#	1053	1073	1304														
BIT5	= 000040	G	1002#	1052	1305															
BIT6	= 000100	G	1001#	1047	1071	1072	1306													
BIT7	= 000200	G	1000#	1051	1066															
BIT8	= 000400	G	999#	1056	1058	1307														
BIT9	= 001000	G	998#	1057	1058	1308														
BOE	= 000400	G	1038#																	
BPRIOR	002212		1105#	1725*																
BRMSG	027524		4814	4825#																
BLF1	003760		1335#	2142	2193	2213	2238	3937	3987	4114	4164	4251	4355	4498	4548					

BUF 2	004760	4673	4723											
BVEC	002214	1336#	2147	2220	2244	3988	4165	4257	4289	4361	4393	4549	4724	
B.BA	002222	1106#	1724*	1775	1820	3819	3824	3879	3884					
B.CS	002220	1109#	1517	2117*										
B.DA	002224	1108#	1519	1982	1984*	1985*	1986	1987*	2116*					
B.MP	002226	1110#	1530	2118*	3948	3951	4125	4128	4509	4512	4684	4687		
CALBCC	002262	1111#	1528	2119*										
CALCRC	015514	1125#	2097	2105	2110	2325*								
CHECK	002304	2087#	3656	3704	3753	3812	3921	4081	4465	4641				
		1134#	3986*	3996	4003*	4022	4163*	4173	4180*	4199	4286*	4296	4303*	4322
		4390*	4400	4407*	4426	4547*	4557	4564*	4583	4722*	4732	4739*	4758	
CHERR	014600	1907#	3934	4111	4495	4670								
CHKOPI	015324	2027#	3669	3717										
CKERLT	014514	1353	1362	1377	1403	1423	1432	1450	1466	1482	1499	1860#		
CLNCOD	014442	1804#												
CNT =	000010	1081#												
COMP	006160	1342#	1926	2041										
CONT	014164	1675	1707	1732#										
CONTIN	014046	1691	1703#											
CRCEND	003064	1264#	4047	4224	4608									
CRDY =	000200	1051#	1811	2376	2761	3019	3021	3306	3413	3467	3524	3580		
CRTIM	006522	1342#	2388											
CSEND	003256	1331#	2897	3042	3092									
CSPAT	003160	1300#	2869	3015	3066									
CSR =	000000	1077#	4797											
CSMSG	027510	4798	4823#											
CSTEST	020200	2874#	2898											
C\$AU =	000052	898#	1851											
C\$AUTO=	000061	898#	1800											
C\$BRK =	000022	898#												
C\$BSEG=	000004	898#	2872	2924	2971	3017	3068	3118	3165	3210	3254	3661	3709	3758
		3817	3924	4084	4468	4645								
C\$BSUB=	000002	898#												
C\$CEFG=	000045	898#												
C\$CLK=	000062	898#												
C\$CLEA=	000012	898#	1825											
C\$CLOS=	000035	898#												
C\$CLP1=	000006	898#	2439	2482	2524	2566	2607	2650	2692	2734	3432	3488	3545	3602
		3616	3668	3677	3716	3725	3769	3782	3860	3877	3933	3936	3947	3962
		3973	3984	4017	4041	4097	4110	4113	4124	4139	4150	4161	4194	4218
		4271	4284	4317	4375	4388	4421	4481	4494	4497	4508	4523	4534	4545
		4578	4602	4656	4669	4672	4683	4698	4709	4720	4753	4777		
C\$CVEC=	000036	898#	1756	1821	2428	2471	2513	2555	2596	2639	2681	2723	3820	3880
C\$DCLN=	000044	898#	1881											
C\$DODU=	000051	898#	1770	1879										
C\$DRPT=	000024	898#												
C\$DU =	000053	898#	1839											
C\$EDIT=	000003	898#	935											
C\$ERDF=	000055	898#	1964	2365	2386	2776	2810	2842	2887	2936	2981	3032	3083	3131
		3176	3220	3265	3321	3355	3387	3427	3441	3483	3498	3540	3554	3597
		3611	3624	3672	3720	3776	3854	3872	3942	3957	3968	3979	3999	4036
		4104	4119	4134	4145	4156	4176	4213	4278	4299	4382	4403	4488	4503
		4518	4529	4540	4560	4597	4663	4678	4693	4704	4715	4735	4772	
C\$ERHR=	000056	898#												
C\$ERRO=	000060	898#												
C\$ERSF=	000054	898#	2434	2477	2519	2561	2602	2645	2687	2729				

SEQ 0093

C\$ERSO=	000057	898#											
C\$ESCA=	000010	898#	2892	2941	2986	3037	3088	3136	3181	3225	3270		
C\$ESEG=	000005	898#	2902	2950	2996	3047	3096	3144	3189	3233	3278	3680	3728
		3890	4044	4221	4605	4780							3785
C\$ESUB=	000003	898#											
C\$ETST=	000001	898#	2442	2485	2527	2569	2610	2653	2695	2737	2783	2818	2850
		2953	2999	3050	3099	3147	3192	3236	3281	3328	3363	3395	3450
		3563	3634	3685	3733	3790	3895	4055	4232	4336	4439	4615	4785
C\$EXIT=	000032	898#											
C\$GETB=	000026	898#											
C\$GETW=	000027	898#											
C\$GMAN=	000043	898#											
C\$GPHR=	000042	898#	1720										
C\$GPLO=	000030	898#											
C\$GPRJ=	000040	898#											
C\$INIT=	000011	898#	1788										
C\$INLP=	000020	898#	1861										
C\$MANI=	000050	898#											
C\$MEM =	000031	898#											
C\$MSG =	000023	898#	1356	1365	1380	1406	1426	1435	1453	1470	1485	1502	
C\$OPEN=	000034	898#											
C\$PNTB=	000014	898#	1375	1390	1401	1420	1448	1463	1480	1497	1512	1525	1535
		1557	1566	1573	1582	1765	4029	4206	4329	4433	4590	4765	1546
C\$PNTF=	000017	898#	1674										
C\$PNTS=	000016	898#											
C\$PNTX=	000015	898#	4014	4191	4314	4418	4575	4750					
C\$QIO =	000377	898#											
C\$RDBU=	000007	898#	1781										
C\$REFG=	000047	898#	1673	1678	1683	1688	1705						
C\$RESE=	000033	898#	1670	2767	2805	2837	3312	3350	3382				
C\$REVI=	000003	898#	934										
C\$RFLA=	000021	898#											
C\$RPT =	000025	898#											
C\$SEFG=	000046	898#											
C\$SPRI=	000041	898#	1809	1816	2759	3304	3761	3772	3836	3866	4089	4100	4264
		4368	4378	4473	4484	4648	4659						4274
C\$SVEC=	000037	898#	1751	1777	2422	2465	2507	2549	2590	2633	2675	2717	3826
C\$TPRI=	000013	898#											3886
DAHS =	000020	1073#											
DATEST	020422	2972#	2992										
DCKMES	006053	1342#	1952										
DELCNT	002354	1154#	2170*	2172*	2173								
DEMES	006021	1342#	1931										
DERFLG	002230	1112#	1907*	1929*	1980								
DERR =	040000	1049#	1927	2762	2764	2878	2880	3023	3025	3074	3076	3307	3309
		3470	3525	3527	3581	3583							3468
DIAGMC=	000000	898#											
DLT =	000000	1086#	4844										
DLTMES	006060	1342#	1956										
DMSG	027624	4845	4863#										
DRBT =	000006	1080#	4807										
DRDY =	000001	1046#	2356	2771	2882	3027	3078	3316	3413	3478	3535	3592	
DRIVE	002216	1107#	1507	1726*	1985	2006	2196						
DRMSG	027544	4808	4829#										
DROP	013610	1611#	1864										
DRPCOD	014504	1832#											
	G												

DRST	=	000010	1067#						
DRTIM		006547	1342#	2367					
DSPCOD		013616	1621#						
DSO	=	000000	1055#						
DS1	=	000400	1056#						
DS2	=	001000	1057#						
DS3	=	001400	1058#						
EF.CON	=	000036	1014#	1704					
EF.NEW	=	000035	1015#	1687					
EF.PWR	=	000034	1016#	1672					
EF.RES	=	000037	1013#	1677					
EF.STA	=	000040	1012#	1682					
ELI	=	000002	1087#	4850					
EMSG		027661	4851	4863#					
EM1		006575	1342#	2436	2604				
EM10		007046	1342#	3944	4121	4505	4680		
EM101		011374	1342#	1562					
EM102		011441	1342#	1569	1578	1922	2034		
EM11		007107	1342#						
EM12		007150	1342#	3959	4136	4520	4695		
EM13		007211	1342#						
EM14		007243	1342#						
EM15		007271	1342#						
EM16		007317	1342#						
EM17		007345	1342#						
EM2		006622	1342#	2479	2647				
EM20		007400	1342#	3970	4147	4531	4706		
EM21		007453	1342#	3981	4158	4542	4717		
EM22		007535	1342#	4001	4178	4562	4737		
EM23		007624	1342#	4038	4215	4599	4774		
EM24		007664	1342#	3778	3856	4106	4280	4384	4490 4665
EM25		007726	1342#	4301					
EM26		007765	1342#	4405					
EM27		010037	1342#	3674					
EM3		006647	1342#	2521	2689				
EM30		010113	1342#	3722					
EM31		010167	1342#	3874					
EM4		006674	1342#	2563	2731				
EM44		010210	1342#	3599					
EM45		010243	1342#	3613					
EM46		010276	1342#	3626					
EM5		006721	1342#	2889					
EM6		006772	1342#	2938					
EM61		010331	1342#	3034					
EM62		010412	1342#	3085					
EM63		010475	1342#	3133					
EM64		010556	1342#	3178					
EM65		010641	1342#	3222					
EM66		010722	1342#	3267					
EM67		011005	1342#	2778	3323				
EM7		007020	1342#	2983					
EM70		011042	1342#	2812	3357				
EM71		011077	1342#	2844	3389				
EM72		011134	1342#	3429					
EM73		011167	1342#	3443					
EM74		011222	1342#	3485					

EM75	011254		1342#	3500												
EM76	011306		1342#	3542												
EM77	011341		1342#	3556												
END	014366		1743	1758	1772#											
ENDDAT	003156		1295#													
ENDPAT	002772		1232#	2945	2991	3140	3185	3229	3274							
ERCOUN	002364		1158#	1694	1713											
ERPOIN	002362		1157#	1713*	1716*	1866*	1867									
ERR =	100000		1048#	2764	2880	3025	3076	3309	3470	3527	3583					
CRRMT	002326		1143#													
ERRVEC	002256		1123#	1749	1755	2420	2427	2463	2470	2505	2512	2547	2554	2588	2595	
			2631	2638	2673	2680	2715	2722								
ERRO	011632	G	1349#	3779	3857	4107	4281	4385	4491	4666						
ERR1	011650	G	1359#	2437	2480	2522	2564	2605	2648	2690	2732					
ERR10	012206	G	1473#	3675	3723											
ERR11	012250	G	1488#	3875												
ERR2	011662	G	1367#	2779	2813	2845	2890	2939	2984	3035	3086	3134	3179	3223	3268	
			3324	3358	3390	3430	3444	3486	3501	3543	3557	3600	3614	3627		
ERR3	011724	G	1384#	4002	4179	4302	4406	4563	4738							
ERR4	012026	G	1411#	3945	3960	3971	3982	4039	4122	4137	4148	4159	4216	4506	4521	
			4532	4543	4600	4681	4696	4707	4718	4775						
ERR5	012074	G	1429#	2368												
ERR6	012106	G	1438#	1967	2389											
ERR7	012150	G	1456#													
EVL =	000004	G	1032#													
EXPMS	006065		1342#	2036												
E\$END =	002100		898#													
E\$LOAD=	000035		898#	947												
E.BA	002234		1114#	1397	1538	2127*	3938	4010	4115	4187	4310	4414	4499	4571	4674	
			4746													
E.CS	002232		1113#	1540	1908	1916	1923	1927	1932	1936	1940	1944	1949	1953	2030	
			2037	2042	2047	2126*										
E.DA	002236		1115#	1396	1552	2128*	3949	4009	4126	4186	4309	4413	4510	4570	4685	
			4745													
E.MP	002240		1116#	1550	2129*	3964	4141	4525	4700							
E.MP1	002242		1117#	1549	2130*	3975	4152	4536	4711							
FIX	015502		1925	1930	1934	1938	1942	1946	1951	1955	1957	1959	2035	2040	2045	
			2050	2054	2056	2069#										
FNDFNC	015316		1992*	1993*	1994	1996*	1999*	2014#								
FRMT1	012674		1509	1586#												
FRMT10	013246		1494	1586#												
FRMT11	013326		1586#	1871												
FRMT12	013367		1586#													
FRMT13	013446		1586#	1762												
FRMT14	013477		1398	1586#	4011	4188	4311	4415	4572	4747						
FRMT2	012734		1522	1543	1586#											
FRMT2A	012753		1532	1586#												
FRMT2B	012766		1554	1586#												
FRMT3	013007		1563	1570	1579	1586#										
FRMT4	013014		1372	1417	1586#											
FRMT5	013140		1586#													
FRMT6	013211		1460	1586#												
FRMT98	013052		1586#	4026	4203	4326	4430	4587	4762							
FRMT99	013135		1387	1445	1477	1586#										
F\$AU =	000015		898#	1846	1850											
F\$AUTO-	000020		898#	1798	1799											

F\$BGN = 000040

898#	907	958	976	1091	1095	1338	1340	1343	1347	1349	1359	1367
1384	1411	1429	1438	1456	1473	1488	1590	1592	1605	1607	1619	1621
1663	1666	1668	1791	1792	1798	1804	1806	1828	1832	1834	1842	1844
1846	1854	1859	2341	2400	2406	2441	2448	2484	2491	2526	2533	2568
2574	2609	2616	2652	2659	2694	2701	2736	2742	2782	2789	2817	2824
2849	2856	2872	2892	2904	2911	2924	2941	2952	2959	2971	2986	2998
3005	3017	3037	3049	3056	3068	3088	3098	3105	3118	3136	3146	3153
3165	3181	3191	3198	3210	3225	3235	3242	3254	3270	3280	3287	3327
3334	3362	3369	3394	3401	3449	3456	3504	3511	3562	3568	3633	3642
3661	3684	3690	3709	3732	3738	3758	3789	3796	3817	3894	3901	3924
4054	4061	4084	4231	4237	4335	4341	4438	4445	4468	4614	4620	4645
4784	4788	4790	4836	4840	4842	4867						

F\$CLEA= 000007

F\$DU = 000016

F\$END = 000041

898#	1806	1824										
898#	1834	1838										
898#	907	958	976	1091	1095	1338	1340	1343	1347	1357	1366	1381
1407	1427	1436	1454	1471	1486	1503	1590	1592	1605	1607	1619	1621
1663	1666	1789	1791	1801	1804	1826	1828	1832	1840	1842	1844	1852
1854	1859	2347	2400	2406	2441	2443	2448	2484	2486	2491	2526	2528
2533	2568	2570	2574	2609	2611	2616	2652	2654	2659	2694	2696	2701
2736	2738	2742	2782	2784	2789	2817	2819	2824	2849	2851	2856	2892
2903	2904	2906	2911	2941	2951	2952	2954	2959	2986	2997	2998	3000
3005	3037	3048	3049	3051	3056	3088	3097	3098	3100	3105	3136	3145
3146	3148	3153	3181	3190	3191	3193	3198	3225	3234	3235	3237	3242
3270	3279	3280	3282	3287	3327	3329	3334	3362	3364	3369	3394	3396
3401	3449	3451	3456	3504	3506	3511	3562	3564	3568	3633	3635	3642
3681	3684	3686	3690	3729	3732	3734	3738	3786	3789	3791	3796	3891
3894	3896	3901	4045	4054	4056	4061	4222	4231	4233	4237	4335	4337
4341	4438	4440	4445	4606	4614	4616	4620	4781	4784	4786	4788	4822
4836	4840	4862	4867									
898#	4790	4820	4848									
898#	1594	1602										
898#	1668	1787										

F\$HARD= 000004

F\$HW = 000013

F\$INIT= 000006

F\$JMP = 000050

F\$MOD = 000000

898#	907	958	976	1091	1095	1338	1340	1343	1347	1590	1592	1605
1607	1619	1621	1663	1666	1791	1804	1828	1832	1842	1844	1854	1859
2400	4788	4836	4840	4867								
898#	1349	1355	1359	1364	1367	1379	1384	1405	1411	1425	1429	1434
1438	1452	1456	1469	1473	1484	1488	1501					
898#	1792	1796										

F\$MSG = 000011

F\$PROT= 000021

F\$PWR = 000017

F\$RPT = 000012

F\$SEG = 000003

898#	2872	2901	2924	2949	2971	2995	3017	3046	3068	3095	3118	3143
3165	3188	3210	3232	3254	3277	3661	3679	3709	3727	3758	3784	3817
3889	3924	4043	4084	4220	4468	4604	4645	4779				
898#	4842	4848	4860									
898#	2341	2345										

F\$SOFT= 000005

F\$SRV = 000010

F\$SUB = 000002

F\$SW = 000014

F\$TEST= 000001

898#	1609	1616										
898#	2406	2441	2448	2484	2491	2526	2533	2568	2574	2609	2616	2652
2659	2694	2701	2736	2742	2782	2789	2817	2824	2849	2856	2904	2911
2952	2959	2998	3005	3049	3056	3098	3105	3146	3153	3191	3198	3235
3242	3280	3287	3327	3334	3362	3369	3394	3401	3449	3456	3504	3511
3562	3568	3633	3642	3684	3690	3732	3738	3789	3796	3894	3901	4054
4061	4231	4237	4335	4341	4438	4445	4614	4620	4784			

G(RCPT 002312

GDAIMP 002322

1137#	2087*	2088	2194
1141#	2215*	2216	2217*

GDCRCA	002314	1138#	2097*	3963	4140	4524	4699							
GDCRCB	002316	1139#	2110*	3974	4151	4535	4710							
GDDAT	002306	1135#	1371	1394	1416	2431*	2474*	2517*	2558*	2599*	2642*	2685*	2726*	2761*
		2764*	2772	2803*	2835*	2875*	2876*	2877	2880*	2883	2926*	2929*	2930	2932
		2973*	2974	2977	3020*	3021*	3025*	3028	3071*	3072*	3076*	3079	3121*	3124*
		3127	3168*	3169*	3172	3213*	3216	3257*	3258*	3261	3306*	3309*	3317	3348*
		3380*	3423*	3437*	3467*	3470*	3471	3479	3494*	3524*	3527*	3528	3536	3550*
		3580*	3583*	3584	3593	3607*	3621*	3937*	3939	3948*	3953*	3954	3963*	3965
		3974*	3976	3990*	3992	4007	4031*	4033	4114*	4116	4125*	4130*	4131	4140*
		4142	4151*	4153	4167*	4169	4184	4208*	4210	4290*	4292	4307	4394*	4396
		4411	4498*	4500	4509*	4514*	4515	4524*	4526	4535*	4537	4551*	4553	4568
		4592*	4594	4673*	4675	4684*	4689*	4690	4699*	4701	4710*	4712	4726*	4728
		4743	4767*	4769										
		1140#	2141*	2144	2212*	2215								
GDDATP	002320	1095#												
GLBDAT	002174	976#												
GLBEQA	002174	1347#												
GLBERR	011632	1859#												
GLBSUB	014514	1340#												
GLBTXT	005760	1066#												
GODRVR	000202	1068#												
GSEBIT	= 000002	1061#	1911	1984										
GSTAT	= 000004	898#												
GSCNTO	= 000200	898#												
G\$DELM	= 000372	898#												
G\$DISP	= 000003	898#												
G\$EXCP	= 000400	898#												
G\$HILI	= 000002	898#												
G\$LOLI	= 000001	898#												
G\$NO	= 000000	898#												
G\$OFFS	= 000400	898#	4793	4797	4802	4807	4813	4844	4850	4856				
G\$OFSI	= 000376	898#	4793	4797	4802	4807	4813	4844	4850	4856				
G\$PRMA	= 000001	898#	4797	4802										
G\$PRMD	= 000002	898#	4807	4813	4850									
G\$PRML	= 000000	898#	4793	4844	4856									
G\$RADA	= 000140	898#												
G\$RADB	= 000000	898#												
G\$RADD	= 000040	898#												
G\$RADL	= 000120	898#	4793	4844	4856									
G\$RADO	= 000020	898#	4797	4802	4807	4813	4850							
G\$XFEP	= 000004	898#	4848											
G\$YES	= 000010	898#	4793	4797	4802	4807	4813	4844	4850	4856				
HCRME	006040	1342#	1943											
HDRBUF	003260	1332#												
HDRIST	015320	1995	2016#											
HINUM	002342	1149#	1692*	1700	1733*	2265	2273	2278*						
HNFMES	006046	1342#	1947	2046										
HOE	= 100000	1045#												
HPTCOD	013572	1592#												
HRDPRM	027434	4788#												
IBE	= 010000	1042#												
IDU	= 000040	1035#												
IER	= 020000	1043#												
INITCO	013732	1666#												
INTEN	= 000100	1047#	1817	1993	2001	3764	3840	4092	4266	4370	4476	4651		
INTFLG	002250	1120#	2342*	2349*	3762*	3773	3780*	3837*	3846	3851	3858*	4090*	4101	4108*
		4261*	4275	4282*	4365*	4379	4386*	4474*	4485	4492*	4649*	4660	4667*	

INTSRV = 016644	1774	2342#	3883										
ISR = 000100 G	1036#												
IXE = 004000 G	1041#												
ISAU = 000041	898#	1846#	1852#										
ISAUTO= 000041	898#	1798#	1801#										
ISCLN = 000041	898#	1806#	1826#										
ISDU = 000041	898#	1834#	1840#										
ISHRD = 000041	4790#	4822#											
ISINIT= 000041	898#	1668#	1789#										
ISMOD = 000041	898#	907#	958#	976#	1091#	1095#	1338#	1340#	1343#	1347#	1590#	1592#	1605#
	1607#	1619#	1621#	1663#	1666#	1791#	1804#	1828#	1832#	1842#	1844#	1854#	1859#
	2400#	4788#	4836#	4840#	4867#								
ISMSG = 000041	898#	1349#	1357#	1359#	1366#	1367#	1381#	1384#	1407#	1411#	1427#	1429#	1436#
	1438#	1454#	1456#	1471#	1473#	1486#	1488#	1503#					
	898#	1792#											
ISPROT= 000040	898#												
ISPTAB= 000041	898#												
ISPR = 000041	898#												
ISRPT = 000041	898#												
ISSEG = 000041	898#	2406	2448	2491	2533	2574	2616	2659	2701	2742	2789	2824	2856
	2872#	2892	2903#	2911	2924#	2941	2951#	2959	2971#	2986	2997#	3005	3017#
	3037	3048#	3056	3068#	3088	3097#	3105	3118#	3136	3145#	3153	3165#	3181
	3190#	3198	3210#	3225	3234#	3242	3254#	3270	3279#	3287	3334	3369	3401
	3456	3511	3568	3642	3661#	3681#	3690	3709#	3729#	3738	3758#	3786#	3796
	3817#	3891#	3901	3924#	4045#	4061	4084#	4222#	4237	4341	4445	4468#	4606#
	4620	4645#	4781#										
ISSETU= 000041	898#												
ISSFT = 000041	4842#	4862#											
ISSRV = 000041	898#	2341#	2347#										
ISSUB = 000041	898#	2406	2448	2491	2533	2574	2616	2659	2701	2742	2789	2824	2856
	2911	2959	3005	3056	3105	3153	3198	3242	3287	3334	3369	3401	3456
	3511	3568	3642	3690	3738	3796	3901	4061	4237	4341	4445	4620	
ISTST = 000041	898#	2406#	2441#	2443#	2448#	2484#	2486#	2491#	2526#	2528#	2533#	2568#	2570#
	2574#	2609#	2611#	2616#	2652#	2654#	2659#	2694#	2696#	2701#	2736#	2738#	2742#
	2782#	2784#	2789#	2817#	2819#	2824#	2849#	2851#	2856#	2904#	2906#	2911#	2952#
	2954#	2959#	2998#	3000#	3005#	3049#	3051#	3056#	3098#	3100#	3105#	3146#	3148#
	3153#	3191#	3193#	3198#	3235#	3237#	3242#	3280#	3282#	3287#	3327#	3329#	3334#
	3362#	3364#	3369#	3394#	3396#	3401#	3449#	3451#	3456#	3504#	3506#	3511#	3562#
	3564#	3568#	3633#	3635#	3642#	3684#	3686#	3690#	3732#	3734#	3738#	3789#	3791#
	3796#	3894#	3896#	3901#	4054#	4056#	4061#	4231#	4233#	4237#	4335#	4337#	4341#
	4438#	4440#	4445#	4614#	4616#	4620#	4784#	4786#					
JSJMP = 000167	898#												
LDCSR = 002252	1121#	1979*	1991*	1992	2001	2006*	2007*	2008	2189*	2195*	2196*	2197*	2198
	3413*	3416											
LDFUN = 016114	2189#	3662	3710	3763	3839	3927	4091	4265	4369	4475	4650		
LDFUNC = 015100	1979#												
LF = 006156	1342#	1966											
LFLG = 002356	1155#	1728*	2168										
LINE1 = 012322	1350	1360	1368	1384	1412	1430	1439	1457	1473	1488	1505#	1767	1876
LINE2 = 012356	1351	1385	1413	1441	1474	1489	1516#						
LINE3 = 012574	1440	1561#											
LINE4 = 012646	1475	1577#											
LOE = 040000 G	1044#												
LONUM = 002344	1150#	1693*	1699	1732*	2241	2264	2271	2277*	4640*				
LOT = 000010 G	1033#												
LTYPF = 000012	1082#	4793											
LTYPMS = 027552	4794	4830#											

.MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 100
 CVRLAC.P11 31-AUG-82 11:25

CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0099

LSACP	002110	G	951#		
LSAPT	002036	G	929#		
LSAU	014510	G	943	1846#	
LSAUT	002070	G	943#		
LSAUTO	014440	G	951	1798#	
LSCCP	002106	G	950#		
LSCLEA	014442	G	950	1806#	
LSCO	002032	G	927#		
LSDEPO	002011	G	918#		
LSDESC	002130	G	946	966#	
LSDESP	002076	G	946#		
LSDEVP	002060	G	939#		
LSDISP	013620	G	930	1624#	
LSDLY	002116	G	954#		
LSDTP	002040	G	930#		
LSDTYP	002034	G	928#		
LSDU	014504	G	944	1834#	
LSDUT	002072	G	944#		
LSDVTY	002122	G	939	963#	
LSEF	002052	G	936#		
LSENV!	002044	G	932#		
SETP	002102	G	948#		
SEXP1	002046	G	933#		
SEXP4	002064	G	941#		
SEXP5	002066	G	942#		
LSHARD	027436	G	921	4790	4791#
LSHIME	002120	G	955#		
LSHPCP	002016	G	921#		
LSHPTP	002022	G	923#		
LSHW	013574	G	923	1594	1595#
LSICP	002104	G	949#		
LSINIT	013752	G	949	1668#	
LSLADP	002026	G	925#		
LSLAST	030520	G	925	4875#	
LSLOAD	002100	G	947#		
LSLUN	002074	G	945#		
LSMREV	002050	G	934#		
LSNAME	002000	G	909#		
LSPRIO	002042	G	931#		
LSPROT	014432	G	952	1792#	
LSPRT	002112	G	952#		
LSREPP	002062	G	940#		
LSREV	002010	G	917#		
LSOFT	027574	G	922	4842	4843#
LSSPC	002056	G	938#		
LSSPCP	002020	G	922#		
LSPTP	002024	G	924#		
LSSTA	002030	G	926#		
LSW	013610	G	924	1609	1610#
LSTEST	002114	G	953#		
LSIML	002014	G	920#		
LSUNIT	002012	G	919#	1712	
L10000	011646		1355#		
L10001	011660		1364#		
L10002	011722		1379#		
L10003	012024		1405#		

[illegible]

.MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 102
 CVRLAC.P11 31-AUG-82 11:25 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0101

MATMES	006324	1342#	3665	3713	3930									
MDHEDR	002000 G	907#												
MERLMT	013612	1612#	1867											
MK	= 000001	1069#												
MSCRLF	006153	1342#	1958	2057										
NODRY	005776	1342#												
NONMES	006122	1342#	2055											
NOPINT	006272	1342#	2017											
NOPMES	006241	1342#	2016											
NOPWR	013744	1676#												
NORES	005760	1342#	1759											
NXM	= 020000	1054#	1932											
NXMMES	006026	1342#	1935											
NXT	014056	1709#	1722	1771										
OPI	= 002000	1050#	1936											
OPIER/	006166	1342#												
OPIMES	006033	1342#	1939	2051										
OPIMN	002336	1147#	1493	3869										
OPIMX	002340	1148#	1492	3828	3867									
OSAPTS=	000000	898#	926											
OSAU	= 000001	898#	904#	943										
OSBGNR=	000000	898#	940											
OSBGNS=	000001	898#	904#	922										
OSDU	= 000001	898#	904#	944										
OSERRT=	000000	898#	948											
OSGNSW=	000001	898#	904#	924										
OSPOIN=	000001	898#	904#	956										
OSSETU=	000000	898#	919	4873										
PATCRC	002774	1236#	3652	3700	3749	3808	3917	4077	4461					
PATDAT	003066	1267#	3653	3701	3750	3809	3918	4078	4462					
PATSAV	002352	1153#	3918*	3920	4046*	4078*	4080	4223*	4462*	4464	4607*			
PFLG	002244	1118#	1779*	1784*										
PNT	= 001000 G	1039#												
PRI	= 002000 G	1040#												
PRIOR	= 000004	1079#	4813											
PRI00	= 000000 G	1028#	1808	3760	3835	4088	4263	4367	4472	4647				
PRI01	= 000040 G	1027#												
PRI02	= 000100 G	1026#												
PRI03	= 000140 G	1025#												
PRI04	= 000200 G	1024#												
PRI05	= 000240 G	1023#												
PRI06	= 000300 G	1022#												
PRI07	= 000340 G	1021#	1815	2758	3303	3771	3865	4099	4273	4377	4483	4658		
RAND	016366	2240	2261#											
RDHDR	= 000010	1063#												
READ	= 000014	1065#												
REST	014124	1718#												
RESTMS	015062	1960#	2005*	2191*										
RHMS	= 000100	1071#												
RLBA	002202	1101#	1737*	2117	2127	2193*	2468*	2474	2636	2642	2799*	2802*	2806	2930*
		2931	3120*	3125*	3126	3167*	3170*	3171	3344*	3347*	3351	3414*	3420	3424
		3473*	3529*	3547	3551	3585*	3604	3608						
RLCS	002200	1100#	1508	1735*	1753	1811	1817*	1986*	1988	2008*	2010*	2116	2126	2198*
		2200*	2356	2376	2425*	2431	2593	2599	2760*	2762	2770	2877*	2878	2881
		3019*	3022*	3023	3026	3070*	3073*	3074	3077	3305*	3307	3315	3416*	3468
		3471*	3477	3525	3528*	3534	3581	3584*	3591					

MAIN. MACV11 30(1046) 31-AUG-82 11:50 PAGE 103
 CVRLAC.P 1 31-AUG-82 11:25

CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 010.

RLDA	002204	1102#	1739*	1983*	2118	2128	2194*	2510*	2517	2678	2685	2834*	2838	2974*
		2976	3212*	3214*	3215	3256*	3259*	3260	3379*	3383	3415*	3434	3438	3472*
		3490	3495	3530*	3586*	3617	3620							
RLMP	002206	1103#	1741*	2119	2129	2130	2190*	2552*	2558	2720	2726	3587*		
SAVCNT	002360	1156#	3985*	3995*	4025	4162*	4172*	4202	4285*	4295*	4325	4389*	4399*	4429
		4546*	4556*	4586	4721*	4731*	4761							
SEEK =	000006	1062#												
SETCMP	016214	2210#	4469											
SETPAT	015752	2139#	3658	3706	3755	3814	3925	4085						
SETRAN	016306	2236#	4643											
SFIPRM	027572 G	4840#												
SIGN =	000004	1070#												
SIMBCC	016464	2093	2101	2106	2297#									
SIZE =	000004	1088#	4856											
SMSG	027650	4857	4863#											
SPTCOD	013606 G	1607#												
START	013776	1680	1685	1692#										
START1	014030	1690	1699#											
START2	014064	1701	1711#											
START3	014106	1710	1715#											
STHS =	000100	1072#												
SVCGBL =	000000	898#	907	909	917	918	919	920	921	922	923	924	925	926
		927	928	929	930	931	932	933	934	936	938	939	940	941
		942	943	944	945	946	947	948	949	950	951	952	953	954
		955	963	966	976	1095	1340	1347	1349	1359	1367	1384	1411	1429
		1438	1456	1473	1488	1592	1595	1607	1610	1621	1624	1666	1668	1792
		1798	1804	1806	1832	1834	1844	1846	1859	4788	4791	4840	4843	4875#
		4876												
SVCINS=	000000	898#	909	910	911	912	913	914	915	916	917	918	919	920
		921	922	923	924	925	926	927	928	929	930	931	932	933
		934	935	936	937	938	939	940	941	942	943	944	945	946
		947	948	949	950	951	952	953	954	955	956	963	964	965
		966	972	973	1356	1357	1365	1366	1370	1371	1372	1373	1374	1375
		1376	1377	1380	1381	1387	1388	1389	1390	1391	1392	1393	1394	1395
		1396	1397	1398	1399	1400	1401	1402	1403	1406	1407	1415	1416	1417
		1418	1419	1420	1421	1422	1426	1427	1435	1436	1445	1446	1447	1448
		1449	1450	1453	1454	1459	1460	1461	1462	1463	1464	1465	1470	1471
		1477	1478	1479	1480	1481	1482	1485	1486	1491	1492	1493	1494	1495
		1496	1497	1498	1499	1502	1503	1506	1507	1508	1509	1510	1511	1512
		1513	1514	1517	1518	1519	1520	1521	1522	1523	1524	1525	1526	1527
		1528	1529	1530	1531	1532	1533	1534	1535	1536	1537	1538	1539	1540
		1541	1542	1543	1544	1545	1546	1547	1548	1549	1550	1551	1552	1553
		1554	1555	1556	1557	1558	1559	1562	1563	1564	1565	1566	1567	1568
		1569	1570	1571	1572	1573	1574	1575	1578	1579	1580	1581	1582	1583
		1584	1594	1595	1609	1610	1623	1624	1625	1626	1627	1628	1629	1630
		1631	1632	1633	1634	1635	1636	1637	1638	1639	1640	1641	1642	1643
		1644	1645	1646	1647	1648	1649	1650	1651	1652	1653	1654	1655	1656
		1657	1658	1659	1660	1661	1670	1671	1672	1673	1674	1675	1676	1677
		1678	1679	1680	1681	1682	1683	1684	1685	1686	1687	1688	1689	1690
		1691	1704	1705	1706	1707	1708	1719	1720	1721	1722	1723	1747	1748
		1749	1750	1751	1752	1753	1755	1756	1757	1761	1762	1763	1764	1765
		1766	1767	1769	1770	1771	1773	1774	1775	1776	1777	1778	1779	1781
		1782	1783	1784	1788	1789	1800	1801	1808	1809	1810	1815	1816	1817
		1820	1821	1822	1825	1826	1839	1840	1851	1852	1861	1862	1863	1864
		1871	1872	1873	1874	1875	1876	1878	1879	1880	1881	1882	1964	1965
		1966	1967	1968	2346	2347	2365	2366	2367	2368	2369	2386	2387	2388

2389	2390	2418	2419	2420	2421	2422	2423	2424	2427	2428	2429	2434
2435	2436	2437	2438	2439	2440	2442	2443	2461	2462	2463	2464	2465
2466	2467	2470	2471	2472	2477	2478	2479	2480	2481	2482	2483	2485
2486	2503	2504	2505	2506	2507	2508	2509	2512	2513	2514	2519	2520
2521	2522	2523	2524	2525	2527	2528	2545	2546	2547	2548	2549	2550
2551	2554	2555	2556	2561	2562	2563	2564	2565	2566	2567	2569	2570
2586	2587	2588	2589	2590	2591	2592	2595	2596	2597	2602	2603	2604
2605	2606	2607	2608	2610	2611	2629	2630	2631	2632	2633	2634	2635
2638	2639	2640	2645	2646	2647	2648	2649	2650	2651	2653	2654	2671
2672	2673	2674	2675	2676	2677	2680	2681	2682	2687	2688	2689	2690
2691	2692	2693	2695	2696	2713	2714	2715	2716	2717	2718	2719	2722
2723	2724	2729	2730	2731	2732	2733	2734	2735	2737	2738	2758	2759
2760	2767	2768	2776	2777	2778	2779	2780	2783	2784	2805	2806	2810
2811	2812	2813	2814	2818	2819	2837	2838	2842	2843	2844	2845	2846
2850	2851	2872	2873	2887	2888	2889	2890	2891	2892	2893	2894	2902
2903	2905	2906	2924	2925	2936	2937	2938	2939	2940	2941	2942	2943
2950	2951	2953	2954	2971	2972	2981	2982	2983	2984	2985	2986	2987
2988	2996	2997	2999	3000	3017	3018	3032	3033	3034	3035	3036	3037
3038	3039	3047	3048	3050	3051	3068	3069	3083	3084	3085	3086	3087
3088	3089	3090	3096	3097	3099	3100	3118	3119	3131	3132	3133	3134
3135	3136	3137	3138	3144	3145	3147	3148	3165	3166	3176	3177	3178
3179	3180	3181	3182	3183	3189	3190	3192	3193	3210	3211	3220	3221
3222	3223	3224	3225	3226	3227	3233	3234	3236	3237	3254	3255	3265
3266	3267	3268	3269	3270	3271	3272	3278	3279	3281	3282	3303	3304
3305	3312	3313	3321	3322	3323	3324	3325	3328	3329	3350	3351	3355
3356	3357	3358	3359	3363	3364	3382	3383	3387	3388	3389	3390	3391
3395	3396	3427	3428	3429	3430	3431	3432	3433	3441	3442	3443	3444
3445	3450	3451	3483	3484	3485	3486	3487	3488	3489	3498	3499	3500
3501	3502	3505	3506	3540	3541	3542	3543	3544	3545	3546	3554	3555
3556	3557	3558	3563	3564	3597	3598	3599	3600	3601	3602	3603	3611
3612	3613	3614	3615	3616	3617	3624	3625	3626	3627	3628	3634	3635
3661	3662	3668	3669	3672	3673	3674	3675	3676	3677	3678	3680	3681
3685	3686	3709	3710	3716	3717	3720	3721	3722	3723	3724	3725	3726
3728	3729	3733	3734	3758	3759	3760	3761	3762	3759	3770	3771	3772
3773	3776	3777	3778	3779	3780	3782	3783	3785	3786	3790	3791	3817
3818	3819	3820	3821	3822	3823	3824	3825	3826	3827	3828	3835	3836
3837	3854	3855	3856	3857	3858	3860	3861	3865	3866	3867	3872	3873
3874	3875	3876	3877	3878	3879	3880	3881	3882	3883	3884	3885	3886
3887	3888	3890	3891	3895	3896	3924	3925	3933	3934	3936	3937	3942
3943	3944	3945	3946	3947	3948	3957	3958	3959	3960	3961	3962	3963
3968	3969	3970	3971	3972	3973	3974	3979	3980	3981	3982	3983	3984
3985	3999	4000	4001	4002	4003	4006	4007	4008	4009	4010	4011	4012
4013	4014	4015	4016	4017	4018	4025	4026	4027	4028	4029	4030	4031
4036	4037	4038	4039	4040	4041	4042	4044	4045	4055	4056	4084	4085
4088	4089	4090	4097	4098	4099	4100	4101	4104	4105	4106	4107	4108
4110	4111	4113	4114	4119	4120	4121	4122	4123	4124	4125	4134	4135
4136	4137	4138	4139	4140	4145	4146	4147	4148	4149	4150	4151	4156
4157	4158	4159	4160	4161	4162	4176	4177	4178	4179	4180	4183	4184
4185	4186	4187	4188	4189	4190	4191	4192	4193	4194	4195	4202	4203
4204	4205	4206	4207	4208	4213	4214	4215	4216	4217	4218	4219	4221
4222	4232	4233	4263	4264	4265	4271	4272	4273	4274	4275	4278	4279
4280	4281	4282	4284	4285	4299	4300	4301	4302	4303	4306	4307	4308
4309	4310	4311	4312	4313	4314	4315	4316	4317	4318	4325	4326	4327
4328	4329	4330	4331	4336	4337	4367	4368	4369	4375	4376	4377	4378
4379	4382	4383	4384	4385	4386	4388	4389	4403	4404	4405	4406	4407
4410	4411	4412	4413	4414	4415	4416	4417	4418	4419	4420	4421	4422

4429	4430	4431	4432	4433	4434	4435	4439	4440	4468	4469	4472	4473
4474	4481	4482	4483	4484	4485	4488	4489	4490	4491	4492	4494	4495
4497	4498	4503	4504	4505	4506	4507	4508	4509	4518	4519	4520	4521
4522	4523	4524	4529	4530	4531	4532	4533	4534	4535	4540	4541	4542
4543	4544	4545	4546	4560	4561	4562	4563	4564	4567	4568	4569	4570
4571	4572	4573	4574	4575	4576	4577	4578	4579	4586	4587	4588	4589
4590	4591	4592	4597	4598	4599	4600	4601	4602	4603	4605	4606	4615
4616	4645	4646	4647	4648	4649	4656	4657	4658	4659	4660	4663	4664
4665	4666	4667	4669	4670	4672	4673	4678	4679	4680	4681	4682	4683
4684	4693	4694	4695	4696	4697	4698	4699	4704	4705	4706	4707	4708
4709	4710	4715	4716	4717	4718	4719	4720	4721	4735	4736	4737	4738
4739	4742	4743	4744	4745	4746	4747	4748	4749	4750	4751	4752	4753
4754	4761	4762	4763	4764	4765	4766	4767	4772	4773	4774	4775	4776
4777	4778	4780	4781	4785	4786	4790	4791	4793	4794	4795	4796	4797
4798	4799	4800	4801	4802	4803	4804	4805	4806	4807	4808	4809	4810
4811	4812	4813	4814	4815	4816	4817	4818	4820	4821	4842	4843	4844
4845	4846	4847	4848	4849	4850	4851	4852	4853	4854	4855	4856	4857
4858	4859	4860	4861	4872	4873	4874	4875					

SVCSUB= 177777
 SVCTAG= 000000

898#	898#	1355	1356	1364	1365	1379	1380	1405	1406	1425	1426	1434
1435	1452	1453	1469	1470	1484	1485	1501	1502	1602	1603	1616	1617
1787	1788	1799	1800	1824	1825	1838	1839	1850	1851	2345	2346	2407
2413	2441	2442	2451	2457	2484	2485	2492	2498	2526	2527	2534	2540
2568	2569	2575	2581	2609	2610	2619	2625	2652	2653	2660	2666	2694
2695	2702	2708	2736	2737	2744	2754	2782	2783	2791	2796	2817	2818
2826	2831	2849	2850	2860	2866	2901	2902	2904	2905	2913	2919	2949
2950	2952	2953	2961	2966	2995	2996	2998	2999	3006	3012	3046	3047
3049	3050	3058	3063	3095	3096	3098	3099	3107	3113	3143	3144	3146
3147	3155	3160	3188	3189	3191	3192	3200	3205	3232	3233	3235	3236
3244	3249	3277	3278	3280	3281	3289	3299	3327	3328	3336	3341	3362
3363	3371	3376	3394	3395	3403	3410	3449	3450	3457	3464	3504	3505
3514	3521	3562	3563	3571	3577	3633	3634	3644	3649	3679	3680	3684
3685	3692	3697	3727	3728	3732	3733	3740	3745	3784	3785	3789	3790
3798	3803	3889	3890	3894	3895	3903	3913	4043	4044	4054	4055	4063
4073	4220	4221	4231	4232	4239	4245	4335	4336	4343	4349	4438	4439
4447	4457	4604	4605	4614	4615	4622	4635	4779	4780	4784	4785	4821
4822	4861	4862										

SVCTST= 177777

898#	2406	2448	2491	2533	2574	2616	2659	2701	2742	2789	2824	2856
2911	2959	3005	3056	3105	3153	3198	3242	3287	3334	3369	3401	3456
3511	3568	3642	3690	3738	3796	3901	4061	4237	4341	4445	4620	

S&LSYM= 010000

898#	1356#	1365#	1380#	1406#	1426#	1435#	1453#	1470#	1485#	1502#	1603#	1617#
1788#	1800#	1825#	1839#	1851#	2346#	2442#	2485#	2527#	2569#	2610#	2653#	2695#
2737#	2783#	2818#	2850#	2872#	2905#	2924#	2953#	2971#	2999#	3017#	3050#	3068#
3099#	3118#	3147#	3165#	3192#	3210#	3236#	3254#	3281#	3328#	3363#	3395#	3450#
3505#	3563#	3634#	3661#	3685#	3709#	3733#	3758#	3790#	3817#	3895#	3924#	4055#
4084#	4232#	4336#	4439#	4468#	4615#	4645#	4785#	4822#	4862#			

TEMP1 002350
 TEMPLO 002346
 TEMP1 002274

1152#	1700*	1733										
1151#	1649*	1732	4639	4640								
1130#	2088*	2089	2091*	2092	2099*	2100	3950*	3951*	3952*	3953	4127*	4128*
4129*	4130	4511*	4512*	4513*	4514	4686*	4687*	4688*	4689			

TEMP2 002264
 TEMP3 002266
 TEMP4 002270
 TEMP5 002272
 TIMSRV 016652
 IMPFNC 002334

1126#	2300*	2323*										
1127#	2301*	2305*										
1128#	2302*	2304	2314*	2316	2321*	2322*	2325					
1129#	2089*	2090*	2091	2098*	2099							
2349#	3823											
1146#	1911	1914	1994*	2192*								

MAIN. MACV11 30(1046) 31-AUG-82 11:50 PAGE 106
 (VRLAC.P11 31-AUG-82 11:25

CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0105

TMPO	002276	1131#	1395	3954*	4008	4171*	4185	4294*	4308	4398*	4412	4555*	4569	4730*
		4744												
TMP1	002300	1132#												
TMP2	002302	1133#												
TRPFLG	002246	1119#	1745*	175	2337*	2416*	2429	2459*	2472	2501*	2514	2543*	2556	2584*
		2597	2627*	2610	2669*	2682	2711*	2724						
TRPHAN	016636	1748	2337#	2419	2462	2504	2546	2587	2630	2672	2714			
T\$ARGC=	000002	909#	910#	911#	912#	913#	914#	1370#	1376	1387#	1391	1393#	1402	1415#
		1421	1445#	1449	1459#	1464	1477#	1481	1491#	1498	1506#	1513	1517#	1526
		1528#	1536	1538#	1547	1549#	1558	1562#	1567	1569#	1574	1578#	1583	1761#
		1766	1871#	1875	4006#	4015	4025#	4030	4183#	4192	4202#	4207	4306#	4315
		4325#	4330	4410#	4419	4429#	4434	4567#	4576	4586#	4591	4742#	4751	4761#
		4766												
T\$CODE=	002130	4793#	4797#	4802#	4807#	4813#	4844#	4848#	4850#	4856#				
T\$ERRN=	000101	898#	1965#	2366#	2387#	2435#	2478#	2520#	2562#	2603#	2646#	2688#	2730#	2777#
		2811#	2843#	2888#	2937#	2982#	3033#	3084#	3132#	3177#	3221#	3266#	3322#	3356#
		3388#	3428#	3442#	3484#	3499#	3541#	3555#	3598#	3612#	3625#	3673#	3721#	3777#
		3855#	3873#	3943#	3958#	3969#	3980#	4000#	4037#	4105#	4120#	4135#	4146#	4157#
		4177#	4214#	4279#	4300#	4383#	4404#	4489#	4504#	4519#	4530#	4541#	4561#	4598#
		4664#	4679#	4694#	4705#	4716#	4736#	4773#						
T\$EXCP=	000000	4797#	4801	4802#	4806	4807#	4812	4813#	4818	4850#	4855			
T\$FLAG=	000040	2892#	2941#	2986#	3037#	3088#	3136#	3181#	3225#	3270#				
T\$GMAN=	000000	898#												
T\$HILI=	177777	4797#	4800	4802#	4805	4807#	4811	4813#	4817	4850#	4854			
T\$LAST=	000001	898#	4873#											
T\$LOLI=	000000	4797#	4799	4802#	4804	4807#	4810	4813#	4816	4850#	4853			
T\$LSYM=	010000	898#	1356	1365	1380	1406	1426	1435	1453	1470	1485	1502	1603	1617
		1788	1800	1825	1839	1851	2346	2442	2485	2527	2569	2610	2653	2695
		2737	2783	2818	2850	2905	2953	2999	3050	3099	3147	3192	3236	3281
		3328	3363	3395	3450	3505	3563	3634	3685	3733	3790	3895	4055	4232
		4336	4439	4615	4785	4822	4862							
T\$LTNO=	000045	4876#												
T\$NEST=	177777	898#	907#	958#	976#	1091#	1095#	1338#	1340#	1343#	1347#	1349#	1355#	1359#
		1364#	1367#	1379#	1384#	1405#	1411#	1425#	1429#	1434#	1438#	1452#	1456#	1469#
		1473#	1484#	1488#	1501#	1590#	1592#	1594#	1602#	1605#	1607#	1609#	1616#	1619#
		1621#	1663#	1666#	1668#	1787#	1791#	1792#	1796#	1798#	1799#	1804#	1806#	1824#
		1828#	1832#	1834#	1838#	1847#	1844#	1846#	1850#	1854#	1859#	2341#	2345#	2400#
		2406#	2441#	2448#	2484#	249	2526#	2533#	2568#	2574#	2609#	2616#	2652#	2659#
		2694#	2701#	2736#	2742#	2782#	2789#	2817#	2824#	2849#	2856#	2872#	2901#	2904#
		2911#	2924#	2949#	2952#	2959#	2971#	2995#	2998#	3005#	3017#	3046#	3049#	3056#
		3068#	3095#	3098#	3105#	3118#	3143#	3146#	3153#	3165#	3188#	3191#	3198#	3210#
		3232#	3235#	3242#	3254#	3277#	3280#	3287#	3327#	3334#	3362#	3369#	3394#	3401#
		3449#	3456#	3504#	3511#	3562#	3568#	3633#	3642#	3661#	3679#	3684#	3690#	3709#
		3727#	3732#	3738#	3758#	3784#	3789#	3796#	3817#	3889#	3894#	3901#	3924#	4043#
		4054#	4061#	4084#	4220#	4231#	4237#	4335#	4341#	4438#	4445#	4468#	4604#	4614#
		4620#	4645#	4779#	4784#	4788#	4790#	4820#	4836#	4840#	4842#	4848	4860#	4867#
T\$NSO =	000000	907#	958	976#	1091	1095#	1338	1340#	1343	1347#	1590	1592#	1605	1607#
		1619	1621#	1663	1666#	1791	1792#	1796	1798#	1799	1804#	1828	1832#	1842
		1844#	1854	1859#	2400	2406#	2441	2448#	2484	2491#	2526	2533#	2568	2574#
		2609	2616#	2652	2659#	2694	2701#	2736	2742#	2782	2789#	2817	2824#	2849
		2856#	2904	2911#	2952	2959#	2998	3005#	3049	3056#	3098	3105#	3146	3153#
		3191	3198#	3235	3242#	3280	3287#	3327	3334#	3362	3369#	3394	3401#	3449
		3456#	3504	3511#	3562	3568#	3633	3642#	3684	3690#	3732	3738#	3789	3796#
		3894	3901#	4054	4061#	4231	4237#	4335	4341#	4438	4445#	4614	4620#	4784
		4788#	4836	4840#	4867									
T\$NS1 =	000005	1349#	1355	1359#	1364	1367#	1379	1384#	1405	1411#	1425	1429#	1434	1438#

CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0106

TSPTNU= 000000
TSSAVL= 177777
TSS EGL= 177777

TSS EKO= 010000

TSSUBN= 000000

TSTAGL= 177777
TSTAGN= 010072

TSTEMP= 000000

TSTEST= 000045

TSTSTM= 177777

1452	1456#	1469	1473#	1484	1488#	1501	1594#	1602	1609#	1616	1668#	1787
1806#	1824	1834#	1838	1846#	1850	2341#	2345	2872#	2901	2924#	2949	2971#
2995	3017#	3046	3068#	3095	3118#	3143	3165#	3188	3210#	3232	3277	3277#
3661#	3679	3709#	3727	3758#	3784	3817#	3889	3924#	4043	4084#	4220	4468#
4604	4645#	4779	4790#	4820	4842#	4848	4860					
898#												
898#												
898#	2872#	2893	2901#	2903	2924#	2942	2949#	2951	2971#	2987	2995#	2997
3017#	3038	3046#	3048	3068#	3089	3095#	3097	3118#	3137	3143#	3145	3165#
3182	3188#	3190	3210#	3226	3232#	3234	3254#	3271	3277#	3279	3661#	3679#
3681	3709#	3727#	3729	3758#	3784#	3786	3817#	3889#	3891	3924#	4043#	4045
4084#	4220#	4222	4468#	4604#	4606	4645#	4779#	4781				
2872#	2893	2901	2924#	2942	2949	2971#	2987	2995	3017#	3038	3046	3068#
3089	3095	3118#	3137	3143	3165#	3182	3188	3210#	3226	3232	3254#	3271
3277	3661#	3679	3709#	3727	3758#	3784	3817#	3889	3924#	4043	4084#	4220
4468#	4604	4645#	4779									
898#	2406#	2448#	2491#	2533#	2574#	2616#	2659#	2701#	2742#	2789#	2824#	2856#
2911#	2959#	3005#	3056#	3105#	3153#	3198#	3242#	3287#	3334#	3369#	3401#	3456#
3511#	3568#	3642#	3690#	3738#	3796#	3901#	4061#	4237#	4341#	4445#	4620#	
898#												
898#	1349#	1359#	1367#	1384#	1411#	1429#	1438#	1456#	1473#	1488#	1594#	1609#
1668#	1792#	1798#	1806#	1834#	1846#	2341#	2406#	2448#	2491#	2533#	2574#	2616#
2659#	2701#	2742#	2789#	2824#	2856#	2911#	2959#	3005#	3056#	3105#	3153#	3198#
3242#	3287#	3334#	3369#	3401#	3456#	3511#	3568#	3642#	3690#	3738#	3796#	3901#
4061#	4237#	4341#	4445#	4620#	4790#	4842#						
958#	1091#	1338#	1343#	1355#	1364#	1379#	1405#	1425#	1434#	1452#	1469#	1484#
1501#	1590#	1602#	1605#	1616#	1619#	1624#	1625#	1626#	1627#	1628#	1629#	1630#
1631#	1632#	1633#	1634#	1635#	1636#	1637#	1638#	1639#	1640#	1641#	1642#	1643#
1644#	1645#	1646#	1647#	1648#	1649#	1650#	1651#	1652#	1653#	1654#	1655#	1656#
1657#	1658#	1659#	1660#	1661#	1663#	1787#	1791#	1796#	1799#	1824#	1828#	1838#
1842#	1850#	1854#	2345#	2400#	2407#	2413#	2441#	2451#	2457#	2484#	2492#	2498#
2526#	2534#	2540#	2568#	2575#	2581#	2609#	2619#	2625#	2652#	2660#	2666#	2694#
2702#	2708#	2736#	2744#	2754#	2782#	2791#	2796#	2817#	2826#	2831#	2849#	2860#
2866#	2892#	2893#	2901#	2904#	2913#	2919#	2941#	2942#	2949#	2952#	2961#	2966#
2986#	2987#	2995#	2998#	3006#	3012#	3037#	3038#	3046#	3049#	3058#	3063#	3088#
3089#	3095#	3098#	3107#	3113#	3136#	3137#	3143#	3146#	3155#	3160#	3181#	3182#
3188#	3191#	3200#	3205#	3225#	3226#	3232#	3235#	3244#	3249#	3270#	3271#	3277#
3280#	3289#	3299#	3327#	3336#	3341#	3362#	3371#	3376#	3394#	3403#	3410#	3449#
3457#	3464#	3504#	3514#	3521#	3562#	3571#	3577#	3633#	3644#	3649#	3679#	3684#
3692#	3697#	3727#	3732#	3740#	3745#	3784#	3789#	3798#	3803#	3889#	3894#	3903#
3913#	4043#	4054#	4063#	4073#	4220#	4231#	4239#	4245#	4335#	4343#	4349#	4438#
4447#	4457#	4604#	4614#	4622#	4635#	4779#	4784#	4793#	4797#	4802#	4807#	4813#
4820#	4836#	4844#	4850#	4856#	4860#	4867#						
898#	2406#	2448#	2491#	2533#	2574#	2616#	2659#	2701#	2742#	2789#	2824#	2856#
2911#	2959#	3005#	3056#	3105#	3153#	3198#	3242#	3287#	3334#	3369#	3401#	3456#
3511#	3568#	3642#	3690#	3738#	3796#	3901#	4061#	4237#	4341#	4445#	4620#	4876
898#	1356	1365	1375	1380	1390	1401	1406	1420	1426	1435	1448	1453
1463	1470	1480	1485	1497	1502	1512	1525	1535	1546	1557	1566	1573
1582	1670	1673	1678	1683	1688	1705	1720	1751	1756	1765	1770	1777
1781	1788	1800	1809	1816	1821	1825	1839	1851	1861	1874	1879	1881
1964	2365	2386	2422	2428	2434	2439	2442	2465	2471	2477	2482	2485
2507	2513	2519	2524	2527	2549	2555	2561	2566	2569	2590	2596	2602
2607	2610	2633	2639	2645	2650	2653	2675	2681	2687	2692	2695	2717
2723	2729	2734	2737	2759	2767	2776	2783	2805	2810	2813	2837	2842
2850	2872	2887	2892	2902	2905	2924	2936	2941	2950	2953	2971	2981
2986	2996	2999	3017	3032	3037	3047	3050	3068	3083	3088	3096	3099

		3118	3131	3136	3144	3147	3165	3176	3181	3189	3192	3210	3220	3225
		3233	3236	3254	3265	3270	3278	3281	3304	3312	3321	3328	3350	3355
		3363	3382	3387	3395	3427	3432	3441	3450	3483	3488	3498	3505	3540
		3545	3554	3563	3597	3602	3611	3616	3624	3634	3661	3668	3672	3677
		3680	3685	3709	3716	3720	3725	3728	3733	3758	3761	3769	3772	3776
		3782	3785	3790	3817	3820	3826	3836	3854	3860	3866	3872	3877	3880
		3886	3890	3895	3924	3933	3936	3942	3947	3957	3962	3968	3973	3979
		3984	3999	4014	4017	4029	4036	4041	4044	4055	4084	4089	4097	4100
		4104	4110	4113	4119	4124	4134	4139	4145	4150	4156	4161	4176	4191
		4194	4206	4213	4218	4221	4232	4264	4271	4274	4278	4284	4299	4314
		4317	4329	4336	4368	4375	4378	4382	4388	4403	4418	4421	4433	4439
		4468	4473	4481	4484	4488	4494	4497	4503	4508	4518	4523	4529	4534
		4540	4545	4560	4575	4578	4590	4597	4602	4605	4615	4645	4648	4656
		4659	4663	4669	4672	4678	4683	4693	4698	4704	4709	4715	4720	4735
		4750	4753	4765	4772	4777	4780	4785						
TSTSTS=	000001	898#	2406#	2448#	2491#	2533#	2574#	2616#	2659#	2701#	2742#	2789#	2824#	2856#
		2911#	2959#	3005#	3056#	3105#	3153#	3198#	3242#	3287#	3334#	3369#	3401#	3456#
		3511#	3568#	3642#	3690#	3738#	3796#	3901#	4061#	4237#	4341#	4445#	4620#	
TSSAU =	010021	1846#	1850											
TSSAUT=	010016	1798#	1799											
TSSCLE=	010017	1806#	1824											
TSSDU =	010020	1834#	1838											
TSSHAR=	010070	4790#	4821											
TSSHW =	010012	1594#	1602											
TSSINI=	010014	1668#	1787											
TSSMSG=	010011	1349#	1355	1359#	1364	1367#	1379	1384#	1405	1411#	1425	1429#	1434	1438#
		1452	1456#	1469	1473#	1484	1488#	1501						
TSSPRO=	010015	1792#												
TSSSEG=	010000	2872#	2892	2901#	2924#	2941	2949#	2971#	2986	2995#	3017#	3037	3046#	3068#
		3088	3095#	3118#	3136	3143#	3165#	3181	3188#	3210#	3225	3232#	3254#	3270
		3277#	3661#	3679#	3709#	3727#	3758#	3784#	3817#	3889#	3924#	4043#	4084#	4220#
		4468#	4604#	4645#	4779#									
TSSSOF =	010071	4842#	4861											
TSSSRV=	010022	2341#	2345											
TSSSW =	010013	1609#	1616											
TSTTES=	010067	2406#	2441	2448#										

MAIN. MACY11 30(1046) 31-AUG-82 11:50 PAGE 109
 CVRLAC.P11 31-AUG-82 11:25 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0108

T20	021246 G	1643	3242#																
T21	021346 G	1644	3287#																
T22	021456 G	1645	3334#																
T23	021530 G	1646	3369#																
T24	021566 G	1647	3401#																
T25	021712 G	1648	3456#																
T26	022052 G	1649	3511#																
T27	022212 G	1650	3568#																
T28	022416 G	1651	3642#																
T29	022526 G	1652	3690#																
T3	017176 G	1626	2491#																
T30	022636 G	1653	3738#																
T31	022772 G	1654	3796#																
T32	023324 G	1655	3901#																
T33	024136 G	1656	4061#																
T34	025014 G	1657	4237#																
T35	025362 G	1658	4341#																
T36	025734 G	1659	4445#																
T37	026612 G	1660	4620#																
T4	017274 G	1627	2533#																
T5	017372 G	1628	2574#																
T6	017466 G	1629	2616#																
T7	017562 G	1630	2659#																
T8	017656 G	1631	2701#																
T9	017752 G	1632	2742#																
UAM =	000200 G	1037#																	
UNITST	002176	1099#	1711*	1715*	1719	1769	1878												
UUT	002174	1098#	1709	1712*	1717*														
VECMG	027535	4803	4827#																
VECT =	000002	1078#	4802																
WDELAY	016032	2165#	2359	2378	3843														
WHY	002330	1144#	1759*	1761															
WRCHK =	000002	1060#	1914																
WRITE -	000012	1064#																	
WTCRDY	016724	2374#	3666	3714	3767	3931	4095	4269	4373	4479	4654								
WTDY	016660	2354#																	
XPOLY	002254	1122#	2310	2321															
X\$ALWA=	000000	898#																	
X\$FALS=	000040	898#	4848																
X\$OFFS=	000400	898#	4848																
X\$TRUE=	000020	898#																	
.	= 030520	900#	1158#	1332#	1335#	1336#	1342#	2893	2942	2987	3038	3089	3137	3182					
		3226	3271	4848	4864#	4868#													

. ABS. 030520 000

ERRORS DETECTED: (1)

(CVRLAC, CVRLAC/CRF:SYM/NL:TOC/SOL=CVRLAC/ML, CVRLAC.P11

RUN-TIME: 30 30 2 SECONDS

RUN-TIME RATIO: 124/62=1.9

CORE USED: 18K (36 PAGES)