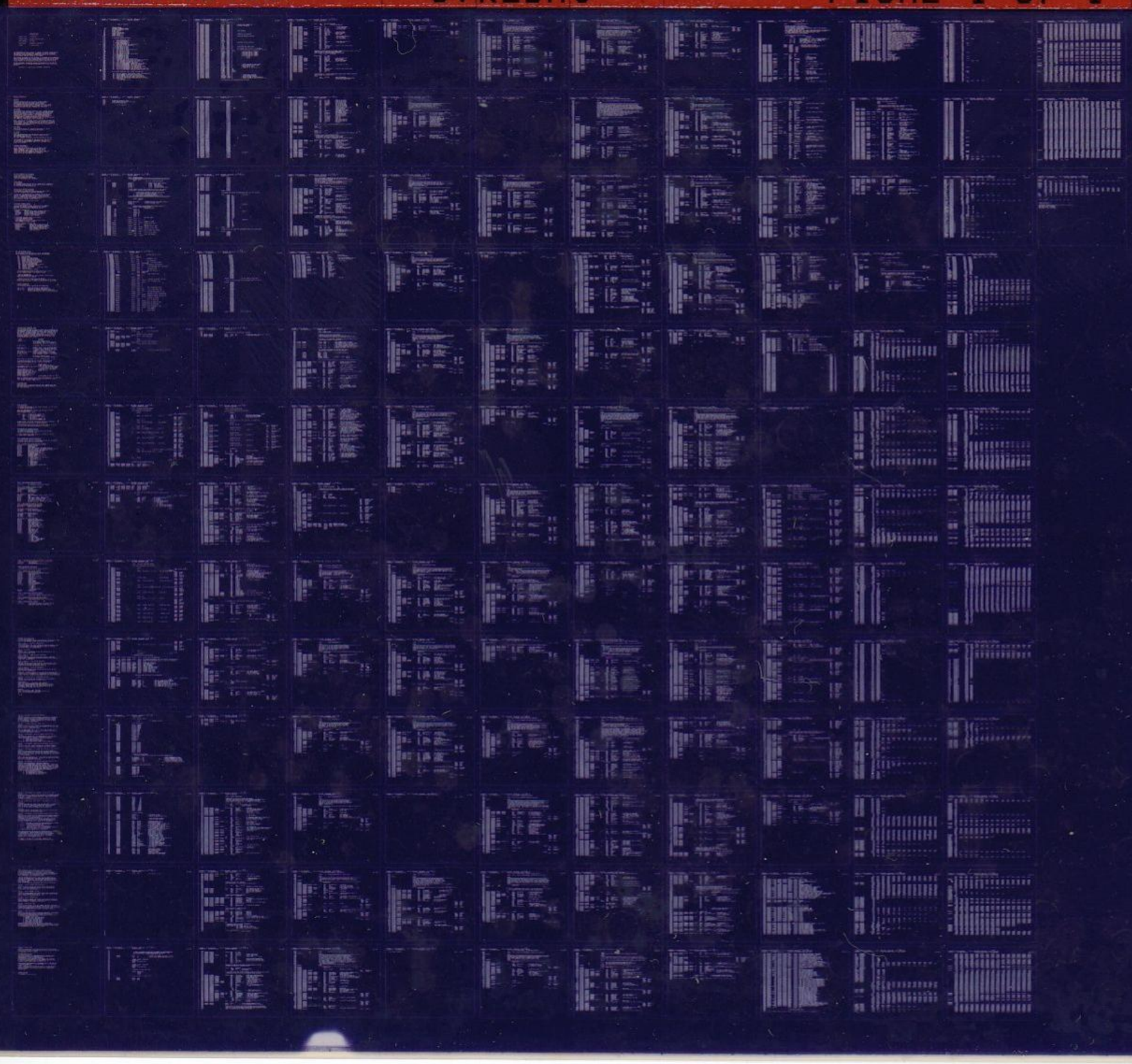
M8061,RLV12 RLV12 DISKLESS CVRLBA0 AH-S827A MC FICHE 1 OF 1 JUL 1982



IDENTIFICATION

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

1.0 GENERAL INFORMATION

1.1 ABSTRACT

THE RLV12 DISKLESS TEST IS AN LSI-11 (PDP-11) BASED PROGRAM THAT WILL TEST THE RLV12, RLV11 AND/OR RL11 DISK CONTROLLERS WITH OR WITHOUT DRIVE ATTACHED.
THE PROGRAM IS AN ADAPTATION OF "CVRLABO RLV11 RL01 DSKLS" RETAINING ALL PREVIOUS TEST CAPABILITIES, AND UPGRADED TO INCLUDE ADDITIONAL TESTING IN SUPPORT OF THE RLV12.

RLV12 MODE

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

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

RLV11 MODE

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

RL11 MODE

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

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

1.2 HARDWARE REQUIREMENTS

LSI-11 (PDP-11) PROCESSOR WITH 16K OR MORE OF CORE.
MEMORY MANAGEMENT, KT11 OR EQUIVALENT (OPTIONAL).
CONSOLE TERMINAL (LA30, LA36, VT52, VT100, ETC.).
XXDP+ LOAD DEVICE AND DIAGNOSTIC MEDIA (RX01, RX02, ETC.)
RLV12, RLV11, OR RL11 CONTROLLER(S) UNDER TEST (1-8).
G5388 RLV12 TEST LOOP MODULE (1-8, OPTIONAL).

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 MISCELLANEOUS

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

2.0 LOADING AND STARTING PROCEDURES

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

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

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

2.1 SUPERVISOR COMMAND SUMMARY

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

THE PRIMARY COMMANDS ARE:

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

THE FOLLOWING SWITCHES APPLY TO THE ABOVE:

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

ADDITIONAL COMMANDS AVAILABLE ARE:

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

2.2 RUN TIME OPTIONS (FLAGS)

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

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

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

```
  STA/FLA:IER:BOE<CR>
```

START THE PROGRAM, RUN ALL TEST IN ORDER, INHIBIT ERROR REPORTS AND RING BELL INSTEAD.

```
  RES/TES:1-5,7,9-29/PAS:10/FLA:HOE:BOE<CR>
```

RESTART AND RUN ALL TESTS EXCEPT 6, 8, AND 30 FOR 10 PASSES. RING THE BELL AND HALT (RETURN TO COMMAND MODE) ON ANY ERROR.

2.3 CONTROL CHARACTERS

THE FOLLOWING CONTROL CHARACTERS ARE RECOGNIZED:

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

2.4 HARD AND SOFT PARAMETER SETUP

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

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

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

2.5 EXECUTION TIME

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

3.0 ERROR REPORTING

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

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

WHERE:

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

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

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

4.0 PERFORMANCE AND PROGRESS REPORTS

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

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

5.0 RLV12 CONTROLLER REGISTER DEFINITION

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

5.1 RLCS -- CONTROL AND STATUS REGISTER (174400)

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

5.2 RLBA -- BUS ADDRESS REGISTER (174402)

BIT<15:0> BUS ADDRESS FOR DMA DATA EXCHANGE.

5.3 RLDA -- DISK ADDRESS REGISTER (174404)

FOR READ/WRITE FUNCTIONS:
BIT<15:7> CYLINDER NUMBER
BIT<6> HEAD SELECT
BIT<5:0> SECTOR NUMBER

FOR SEEK/STATUS FUNCTIONS:
BIT<15:7> SEEK DIFFERENCE (SEEK), UNUSED (STATUS)
BIT<6:5> ZERO
BIT<4> HEAD SELECT (SEEK), ZERO (STATUS).
BIT<3> ZERO (SEEK), RESET (STATUS)
BIT<2> SEEK DIRECTION (SEEK), ZERO (STATUS)
BIT<1> 0 = SEEK, 1 = GET STATUS
BIT<0> MARKER, MUST BE 1 FOR EITHER FUNCTION.

5.4 RLMP -- MULTIPURPOSE REGISTER (174406)

WRITE BEFORE READ/WRITE DATA FUNCTIONS:
BIT<15:0> NEGATIVE WORD COUNT

READ AFTER MAINTENANCE FUNCTION:
1ST WORD: CRC OF INITIAL (DA)+3
2ND WORD: CRC OF CRC OF INITIAL (DA)+4

READ AFTER READ-HEADER FUNCTION:
1ST WORD: DISK ADDRESS.
2ND WORD: ZERO.
3RD WORD: HEADER CRC.

READ AFTER GET STATUS FUNCTION:
BIT<15> WRITE DATA ERROR (WDE)
BIT<14> CURRENT HEAD ERROR (CHE)
BIT<13> WRITE LOCK (WL)
BIT<12> SEEK TIME-OUT (SKTO)
BIT<11> SPIN ERROR (SPE)
BIT<10> WRITE GATE ERROR (WGE)
BIT<9> VOLUME CHECK (VC)
BIT<8> DRIVE SELECT ERROR (DSE)
BIT<7> ZERO
BIT<6> HEAD SELECT (HS)
BIT<5> COVER OPEN (CO)
BIT<4> HEADS OUT (HO)
BIT<3> BRUSHES HOME (BH)
BIT<2:0> DRIVE STATE:
0 = LOAD
1 = SPIN UP
2 = BRUSH CYCLE
3 = LOAD HEADS
4 = SEEK - TRACK COUNTING
5 = SEEK - LINEAR MODE
6 = UNLOAD HEADS
7 = SPIN DOWN

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

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

6.0 G5388 TLM REGISTER DEFINITION

THE G5388 TLM UTILIZES THE FOLLOWING REGISTERS:

6.1 TCSR -- CONTROL AND STATUS (160010)

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

6.2 TSKGS -- SEEK/STATUS REGISTER (160012)

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

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

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

6.4 TPROM -- PROM REGISTER (160016)

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

7.0 HARDWARE TEST DESCRIPTION

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

7.1 TESTS 1 THRU 5 -- REGISTER ADDRESSABILTY.

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

7.2 TEST 6 -- BUS RESET.

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

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

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

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

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

7.5 TESTS 15 THRU 19 -- REGISTER UNIQUENESS.

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

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

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

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

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

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

7.7 TEST 21 -- INTERRUPT ON FUNCTION COMPLETE.

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

7.8 TEST 22 -- INTERRUPT PRIORITY LEVEL TEST.

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

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

7.9 TEST 23 -- MAINTENANCE, FORCE OPI ERROR (RLV12/11 ONLY).

THIS TEST WILL VERIFY THAT THE CONTROLLER WILL SET THE OPI (OPERATION INCOMPLETE) BIT ON A MAINTENANCE FUNCTION IF THE INITIAL WORD COUNT IS INCORRECT. WE'LL DO 2 PASSES.

1. WORD COUNT LESS THAN 510.
2. WORD COUNT GREATER THAN 511.

BOTH CASES SHOULD CAUSE AN OPI ERROR IN THE CONTROLLER.

7.10 TEST 24 -- MAINTENANCE, FORCE OPI AND INTERRUPT (RLV12/11 ONLY).

SIMILAR TO THE LAST, EXCEPT EXECUTE WITH INTERRUPT ENABLED. EXPECT OPI ERROR TO CAUSE AN INTERRUPT THRU THE RL VECTOR.

7.11 TEST 25 -- MAINTENANCE, OPI TIMING TEST (RLV12/11 ON LSI ONLY).

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

NOTE: THIS TEST EMPLOYS A SOFT TIMER THAT HAS BEEN CALIBRATED FOR LSI11/2 AND /23 CPUS ONLY. IF YOUR CPU IS NOT ONE OF THESE THIS TEST IS AUTOMATICALLY BYPASSED.

7.12 TEST 26 -- MAINTENANCE, FIFO AND DMA TRANSFER (RLV12/11 ONLY).

VERIFIES THAT "MAINTENANCE" FUNCTIONS CORRECTLY.
EXECUTE A MAINTENANCE MODE FUNCTION IN FLAG MODE. VERIFY THE FINAL BA, DA, AND BAE REGISTERS ARE CORRECT. CHECK THE SERIAL WRITE/READ DATA PATHS BY READING THE TWO MAINTENANCE CRC WORDS (CRC OF DA+3, AND CRC OF CRC OF DA+4) FROM THE FIFO VIA THE MP REGISTER. VERIFY THAT 256 WORDS WERE TRANSFERED FROM MEMORY TO THE FIFO, AND 255 WORDS FROM FIFO BACK TO MEMORY.
REPEAT 57 TIMES USING VARIOUS DATA.

1. 28 DIFFERENT DATA PATTERNS.
2. THE COMPLIMENT OF THOSE PATTERNS.
3. ONE RANDOM 256 WORD PATTERN.

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

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

7.14 TEST 28 -- MAINTENANCE, BANK SELECT 7 AND NXM (RLV12/11 ONLY).

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

FOR RLV11:
SINCE RLV11 DOESN'T ASSERT BBS7, WE'LL EXECUTE THIS TEST ONLY IF MEMORY SIZE IS LESS THAN 124K WORDS.

7.15 TEST 29 -- MAINTENANCE, EXTENDED MEMORY ACCESS (RLV12/11 ONLY).

THIS TEST WILL VERIFY THAT THE CONTROLLER CAN ADDRESS EXTENDED MEMORY UP TO 124K (18 BIT) OR 2044K (22 BIT) DEPENDING UPON CONTROLLER TYPE AND CONFIGURATION.
IT REQUIRES THE SERVICES OF THE MEMORY MANAGEMENT FACILITY (KT11 OR EQUIVALENT IN 11/23) TO ESTABLISH AND VERIFY DATA BUFFERS IN MEMORY FROM 32K TO 2044K, IN 4K INCREMENTS.

1. SETUP -- USING MMU, INITIALIZE A WRITE BUFFER WITH A RANDOM DATA PATTERN, AND CLEAR THE READ BUFFER.
2. EXECUTE -- RLV12/11 MAINTENANCE MODE. WRITE BUFFER => FIFO => READ BUFFER.
3. VERIFY -- USING MMU, VERIFY THAT THE READ BUFFER RECEIVED A COPY OF THE DATA IN THE WRITE BUFFER.
4. INCREMENT BUFFER ADDRESS BY 4K, AND REPEAT.

NOTE THAT THE ENTIRE 18 (OR 22) BIT ADDRESS SPACE IS TESTED ON 4K BOUNDARIES, WHETHER MEMORY EXISTS THERE OR NOT.
IF MEMORY EXISTS, VERIFY THE DATA EXCHANGE AS DESCRIBED.
IF NOT, VERIFY THAT THE CONTROLLER GOT A "NXM ERROR".
CONTINUE ALL THE ABOVE UNTIL THE I/O PAGE IS REACHED AT 760000 (18 BIT) OR 17760000 (22 BIT).

OF COURSE, IF AN MMU ISN'T AVAILABLE, THEN ALL THIS IS ACADEMIC, AND WE'LL JUST FALL THRU TO THE NEXT TEST.

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

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

FROM THE CONTROLLERS POINT OF VIEW, THE TLM LOOKS LIKE A 6 SECTOR RL01/RL02 DISK. VARIOUS HEADER AND DATA PATTERNS ARE STORED IN PROM ON THE TLM, AND ARE ACCESSABLE THRU THE EXECUTION OF "NORMAL" READ/WRITE AND STATUS FUNCTIONS IN THE RLV12 CONTROLLER.
REFER TO THE G5388 TLM ENGINEERING SPEC FOR FURTHER DETAILS.

7.16.1 TEST 30 -- STATIC CONTROL AND STATUS BITS.

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

7.16.2 TEST 31 -- DRIVE COMMAND, STATUS.

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

7.16.3 TEST 32 -- DRIVE COMMAND, SEEK.

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

7.16.4 TEST 33 -- WRITE DATA PATH.

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

7.16.5 TEST 34 -- READ DATA PATHS.

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

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

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

7.16.6 TEST 35 -- WRITE CHECK.

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

7.16.7 TLM PROM DUMP UTILITY.

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

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

8.0 PROGRAM LISTING

THE PROGRAM LISTING FOLLOWS:

TABLE OF CONTENTS

```
                        .SBTTL  PROGRAM HEADER
                        ;*******************************************************************
                                .MCALL  SVC
000000'                         SVC             ; INITIALIZE SUPERVISOR MACROS

        000001          SVCGBL= 1               ; LIST GLOBAL TAGS AT RIGHT MARGIN.
        000001          SVCTST= 1               ;      DITTO   TEST TAGS.
        000001          SVCSUB= 1               ;      DITTO   SUBTEST TAGS.
        000001          SVCTAG= 1               ;      DITTO   ANY OTHER TAGS.
        000001          SVCINS= 1               ;      DITTO   INSTRUCTIONS AND DATA.
                        ;
                        ; THESE SYMBOLS CONTROL THE LISTING FIELD OF ALL SVC MACRO
                        ; EXPANSIONS.  YOU MAY CHANGE THEM AT ANY TIME OR PLACE.
                        ;
                        ;        1 = RIGHT-JUSTIFY (MAKES IT EASY TO DISTINGUISH
                        ;            SVC'S MACRO CODE FROM YOUR OWN).
                        ;        0 = LEFT-JUSTIFY (ALIGN IN A NORMAL FASHION).
                        ;       -1 = DON'T LIST THE EXPANSIONS AT ALL.
                        ;*******************************************************************

                                .ENABL  ABS,AMA
        002000                  .=      2000

        000100          PRGSIZ= ^H<L$LAST>      ; PROGRAM SIZE IN 1/8 K UNITS (OCTAL).
        000000          SVCGBL= 0               ; ALIGN THE HEADER STUFF.
        000000          SVCINS= 0

002000                          POINTER BGNSFT,BGNSW,BGNDU,BGNAU,BGNRPT,BGNSETUP
002000                          HEADER  CVRLB,A,0,30.,0,340
(4) 002000              L$NAME::                ;DIAGNOSTIC NAME
(4) 002000      103             .ASCII  /C/
(4) 002001      126             .ASCII  /V/
(4) 002002      122             .ASCII  /R/
(4) 002003      114             .ASCII  /L/
(4) 002004      102             .ASCII  /B/
(6) 002005      000             .BYTE   0
(6) 002006      000             .BYTE   0
(5) 002007      000             .BYTE   0
(5) 002010              L$REV::                 ;REVISION LEVEL
(4) 002010      101             .ASCII  /A/
(5) 002011              L$DEPO::                ;0
(4) 002011      060             .ASCII  /0/
(5) 002012              L$UNIT::                ;NUMBER OF UNITS
(4) 002012  000001              .WORD   T$PTHV
(5) 002014              L$TIML::                ;LONGEST TEST TIME
(4) 002014  000036              .WORD   30.
(5) 002016              L$HPCP::                ;POINTER TO H.W. QUES.
(4) 002016  002172              .WORD   L$HARD
(5) 002020              L$SPCP::                ;POINTER TO S.W. QUES.
(4) 002020  002412              .WORD   L$SOFT
(5) 002022              L$HPTP::                ;PTR. TO DEF. H.W. PTABLE
(4) 002022  002376              .WORD   L$HW
(5) 002024              L$SPTP::                ;PTR. TO S.W. PTABLE
(4) 002024  003304              .WORD   L$SW
(5) 002026              L$LADP::                ;DIAG. END ADDRESS
(4) 002026  040344              .WORD   L$LAST
```

```
(5)  002030                    L$STA::                 ;RESERVED FOR APT STATS
(4)  002030  000000                    .WORD   0
(5)  002032                    L$CO::
(4)  002032  000000                    .WORD   0
(5)  002034                    L$DTYP::                ;DIAGNOSTIC TYPE
(4)  002034  000000                    .WORD   0
(5)  002036                    L$APT::                 ;APT EXPANSION
(4)  002036  000000                    .WORD   0
(5)  002040                    L$DTP::                 ;PTR. TO DISPATCH TABLE
(4)  002040  026052                    .WORD   L$DISPATCH
(5)  002042                    L$PRIO::                ;DIAGNOSTIC RUN PRIORITY
(4)  002042  000340                    .WORD   340
(5)  002044                    L$ENVI::                ;FLAGS DESCRIBE HOW IT WAS SETUP
(4)  002044  000000                    .WORD   0
(5)  002046                    L$EXP1::                ;EXPANSION WORD
(4)  002046  000000                    .WORD   0
(5)  002050                    L$MREV::                ;SVC REV AND EDIT #
(4)  002050     003                    .BYTE   C$REVISION
(3)  002051     003                    .BYTE   C$EDIT
(5)  002052                    L$EF::                  ;DIAG. EVENT FLAGS
(4)  002052  000000                    .WORD   0
(5)  002054  000000                    .WORD   0
(5)  002056                    L$SPC::
(4)  002056  000000                    .WORD   0
(5)  002060                    L$DEVP::                ; POINTER TO DEVICE TYPE LIST
(4)  002060  002142                    .WORD   L$DVTYP
(5)  002062                    L$REPP::                ;PTR. TO REPORT CODE
(4)  002062  011644                    .WORD   L$RPT
(5)  002064                    L$EXP4::
(4)  002064  000000                    .WORD   0
(5)  002066                    L$EXP5::
(4)  002066  000000                    .WORD   0
(5)  002070                    L$AUT::                 ;PTR. TO ADD UNIT CODE
(4)  002070  007432                    .WORD   L$AU
(5)  002072                    L$DUT::                 ;PTR. TO DROP UNIT CODE
(4)  002072  007350                    .WORD   L$DU
(5)  002074                    L$LUN::                 ;LUN FOR EXERCISERS TO FILL
(4)  002074  000000                    .WORD   0
(5)  002076                    L$DESP::                ;POINTER TO DIAG. DESCRIPTION
(4)  002076  002122                    .WORD   L$DESC
(5)  002100                    L$LOAD::                ;GENERATE SPECIAL AUTOLOAD EMT
(4)  002100  104035                    EMT     E$LOAD
(5)  002102                    L$ETP::                 ;POINTER TO ERRTBL
(4)  002102  000000                    .WORD   0
(5)  002104                    L$ICP::                 ;PTR. TO INIT CODE
(4)  002104  006374                    .WORD   L$INIT
(5)  002106                    L$CCP::                 ;PTR. TO CLEAN-UP CODE
(4)  002106  007272                    .WORD   L$CLEAN
(5)  002110                    L$ACP::                 ;PTR. TO AUTO CODE
(4)  002110  007214                    .WORD   L$AUTO
(5)  002112                    L$PRT::                 ;PTR. TO PROTECT TABLE
(4)  002112  006366                    .WORD   L$PROT
(5)  002114                    L$TEST::                ;TEST NUMBER
(4)  002114  000000                    .WORD   0
(5)  002116                    L$DLY::                 ;DELAY COUNT
(4)  002116  000000                    .WORD   0
```

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

```
                        .SBTTL  HARDWARE PARAMETER CODING
                        ;
                        ; GET PARAMETERS FROM OPERATOR.
                        ;
002170                          BGNHRD
(3) 002170 000036                                                       L$SHARD::.WORD L10000-L$SHARD/2
(3) 002172
002172
(4) 002172 004130               GPRML   RL1,10,BIT1,YES         ; RLV12 ??
(4) 002174 002266                                                               .WORD   T$CODE
(4) 002176 000002                                                               .WORD   RL1
                                                                                .WORD   BIT1
002200                          XFERF   1$
(5) 002200 005044                                                               .WORD   T$CODE
002202                          GPRML   RL1A,10,BIT0,YES        ; BAE ENABLED ??
(4) 002202 004130                                                               .WORD   T$CODE
(4) 002204 002276                                                               .WORD   RL1A
(4) 002206 000001                                                               .WORD   BIT0
002210                          XFER    2$
(5) 002210 004004                                                               .WORD   T$CODE
002212          1$:     GPRML   RL2,10,BIT0,YES         ; RLV11 ??
(4) 002212 004130                                                               .WORD   T$CODE
(4) 002214 002313                                                               .WORD   RL2
(4) 002216 000001                                                               .WORD   BIT0
002220          2$:     GPRMA   CSRA,0,0,160000,177776,YES ; CSR.
(4) 002220 000031                                                               .WORD   T$CODE
(4) 002222 002323                                                               .WORD   CSRA
(4) 002224 160000                                                               .WORD   T$LOLIM
(4) 002226 177776                                                               .WORD   T$HILIM
002230                          GPRMA   VECA,2,0,0,776,YES      ; VECTOR.
(4) 002230 001031                                                               .WORD   T$CODE
(4) 002232 002340                                                               .WORD   VECA
(4) 002234 000000                                                               .WORD   T$LOLIM
(4) 002236 000776                                                               .WORD   T$HILIM
002240                          GPRMD   BRL,4,0,340,0,7,YES     ; BR LEVEL.
(4) 002240 002032                                                               .WORD   T$CODE
(4) 002242 002351                                                               .WORD   BRL
(4) 002244 000340                                                               .WORD   340
(4) 002246 000000                                                               .WORD   T$LOLIM
(4) 002250 000007                                                               .WORD   T$HILIM
002252                          XFER    3$              ; DRIVE IS ALWAYS 0 IN DISKLESS.
(5) 002252 006004                                                               .WORD   T$CODE
002254                          GPRMD   DRN,6,0,3400,0,7,YES    ; DRIVE.
(4) 002254 003032                                                               .WORD   T$CODE
(4) 002256 002363                                                               .WORD   DRN
(4) 002260 003400                                                               .WORD   3400
(4) 002262 000000                                                               .WORD   T$LOLIM
(4) 002264 000007                                                               .WORD   T$HILIM
002266          3$:     ENDHRD
(2)                                                                             .EVEN
(3) 002266                                                              L10000:
                        ;
                        ; THESE ARE THE HARDWARE QUERIES...
                        ;
046122 030526 004462    RL1:    .ASCIZ  'RLV12          '
040502 020105 047105    RL1A:   .ASCIZ  'BAE ENABLED    '
   122 053114 030461    RL2:    .ASCIZ  'RLV11          '
```

```
                                                                                                    H 2
     73  002323      103  051123  040440  CSRA:   .ASCIZ  'CSR ADDRESS    '
     74  002340  042526  052103  051117  VECA:   .ASCIZ  'VECTOR         '
     75  002351      102  020122  042514  BRL:    .ASCIZ  'BR LEVEL       '
     76  002363      104  044522  042526  DRN:    .ASCIZ  'DRIVE          '          ; UNUSED.
     77          002374                          .EVEN
     78                                  ;
     79                                  ;...AND THESE ARE THE SINGLE UNIT DEFAULTS (HARD P-TABLE).
     80                                  ;
     81  002374                                  BGNHW
(3)  002374  000005                                                                 .WORD   L10001-L$HW/2
(3)  002376                                                                 L$HW::
     82  002376  174400                        .WORD   174400          ; (0) CSR BUS ADDRESS.
     83  002400  000160                        .WORD   160             ; (2) VECTOR
     84  002402  000200                        .WORD   PRI04           ; (4) PRIORITY
     85  002404  000000                        .WORD   0               ; (6) DRIVE (BITS 8,9,10)
     86  002406  000003                        .WORD   3               ; (10) CONTROLLER TYPE...
     87                                                             ;.....0 = RL11...
     88                                                             ;.....1 = RLV11...
     89                                                             ;.....2 = RLV12, BAE DISABLED...
     90                                                             ;.....3 = RLV12, BAE ENABLED.
     91  002410                                  ENDHW
(3)  002410                                                                 L10001:
```

```
                                                                    I 2
                        .SBTTL  SOFTWARE PARAMETER CODING
                        ;
                        ; GET SOFT PARAMETERS FROM OPERATOR.
                        ;
 002410                         BGNSFT
(3) 002410 000054                                                       .WORD L10002-L$SOFT/2
(3) 002412                                                      L$SOFT::
 002412                         GPRMD   ADLIM,0,D,-1,0,-1,YES   ; GET AUTO-DROP ERROR LIMIT.
(4) 002412 000052                                                       .WORD   T$CODE
(4) 002414 002542                                                       .WORD   ADLIM
(4) 002416 177777                                                       .WORD   -1
(4) 002420 000000                                                       .WORD   T$LOLIM
(4) 002422 177777                                                       .WORD   T$HILIM
 002424                         DISPLAY TLMBP                   ; BYPASS MESSAGE.
(4) 002424 000003                                                       .WORD   T$CODE
(4) 002426 002575                                                       .WORD   TLMBP
 002430                         GPRML   TLMIN,12,BIT0,YES       ; TLM ON THIS UNIT ??
(4) 002430 005130                                                       .WORD   T$CODE
(4) 002432 002751                                                       .WORD   TLMIN
(4) 002434 000001                                                       .WORD   BIT0
 002436                         XFERF   1$
(5) 002436 042044                                                       .WORD   T$CODE
 002440                         GPRMA   TCSU0,14,0,160000,177776,YES ; TLM CSR FOR UNIT 0.
(4) 002440 006031                                                       .WORD   T$CODE
(4) 002442 002776                                                       .WORD   TCSU0
(4) 002444 160000                                                       .WORD   T$LOLIM
(4) 002446 177776                                                       .WORD   T$HILIM
 002450                         GPRML   PRMDMP,2,BIT0,YES       ; PROM DUMP ??
(4) 002450 001130                                                       .WORD   T$CODE
(4) 002452 003024                                                       .WORD   PRMDMP
(4) 002454 000001                                                       .WORD   BIT0
 002456                         XFERT   1$
(5) 002456 032024                                                       .WORD   T$CODE
 002460                         GPRMD   PRM,4,0,-1,0,-1,YES     ; PROM ID.
(4) 002460 002032                                                       .WORD   T$CODE
(4) 002462 003047                                                       .WORD   PRM
(4) 002464 177777                                                       .WORD   -1
(4) 002466 000000                                                       .WORD   T$LOLIM
(4) 002470 177777                                                       .WORD   T$HILIM
 002472                         GPRML   INHMXP,6,BIT15,YES      ; INHIBIT MAX PEAK...
(4) 002472 003130                                                       .WORD   T$CODE
(4) 002474 003070                                                       .WORD   INHMXP
(4) 002476 100000                                                       .WORD   BIT15
 002500                         GPRML   INHMNP,6,BIT14,YES      ;...MIN PEAK...
(4) 002500 003130                                                       .WORD   T$CODE
(4) 002502 003120                                                       .WORD   INHMNP
(4) 002504 040000                                                       .WORD   BIT14
 002506                         GPRML   INHNC,6,BIT2,YES        ;...NOMINAL CLOCK...
(4) 002506 003130                                                       .WORD   T$CODE
(4) 002510 003150                                                       .WORD   INHNC
(4) 002512 000004                                                       .WORD   BIT2
 002514                         GPRML   INHFC,6,BIT1,YES        ;...FAST CLOCK...
(4) 002514 003130                                                       .WORD   T$CODE
(4) 002516 003177                                                       .WORD   INHFC
(4) 002520 000002                                                       .WORD   BIT1
 002522                         GPRML   INHSC,6,BIT0,YES        ;...SLOW CLOCK.
```

```
(4)  002522  003130                                                        .WORD   T$CODE
(4)  002524  003223                                                        .WORD   INHSC
(4)  002526  000001                                                        .WORD   BITO
111  002530                      GPRMD   SEC1,10,0,7,0,6,YES     ; SINGLE SECTOR OPTION.
(4)  002530  004032                                                        .WORD   T$CODE
(4)  002532  003247                                                        .WORD   SEC1
(4)  002534  000007                                                        .WORD   7
(4)  002536  000000                                                        .WORD   T$LOLIM
(4)  002540  000006                                                        .WORD   T$HILIM
112  002542                 1$:     ENDSFT
(2)                                                                        .EVEN
(3)  002542                                                        L10002:
113                                 ;
114                                 ; THESE ARE THE SOFTWARE QUERIES...
115                                 ;
116  002542  051105  047522  020122  ADLIM:  .ASCIZ  'ERROR LIMIT FOR AUTO-DROP '
117
118  002575    10$:  046114  051040  TLMBP:  .ASCIZ  'ALL REMAINING QUERIES ARE FOR OPTIONAL (MANUFACTURING)'
119  002664  032507  034063  020070          .ASCIZ  'G5388 TEST-LOOP-MODULE SET-UP.  USE <^Z> TO BYPASS.'
120  002750     000                          .BYTE   0
121  002751     107  031465  034070  TLMIN:  .ASCIZ  'G5388 TLM INSTALLED    '
122  002776  051503  020122  042101  TCSU0:  .ASCIZ  'CSR ADDRESS (UNIT 0)   '
123  003024  052504  050115  050040  PRMDMP: .ASCIZ  'DUMP PROM ON TTY:      '
124  003047     120  047522  020115  PRM:    .ASCIZ  'PROM ID NUMBER         '
125  003070  047111  044510  044502  INHMXP: .ASCIZ  'INHIBIT MAX PEAK SHIFT '
126  003120  047111  044510  044502  INHMNP: .ASCIZ  'INHIBIT MIN PEAK SHIFT '
127  003150  047111  044510  044502  INHNC:  .ASCIZ  'INHIBIT NOMINAL CLOCK  '
128  003177     111  044116  041111  INHFC:  .ASCIZ  'INHIBIT FAST CLOCK     '
129  003223     111  044116  041111  INHSC:  .ASCIZ  'INHIBIT SLOW CLOCK     '
130  003247     123  041505  047524  SEC1:   .ASCIZ  'SECTOR NUMBER (0=USE ALL)'
131          003302                  .EVEN
132                                 ;
133                                 ;... AND THESE ARE THE DEFAULT SOFT PARAMETERS.
134                                 ;
135  003302                             BGNSW
(3)  003302  000007                                                        .WORD   L10003-L$SW/2
(3)  003304                                                        L$SW::
136  003304  000000         ERRLMT: .WORD   0               ; (00) NZ = DROP AFTER N ERRORS.
137  003306  000000         PDSW:   .WORD   0               ; (02) PROM DUMP SWITCH (BIT0).
138  003310  000401         PROMID: .WORD   401             ; (04) CURRENT PROM SET ID.
139  003312  000000         MPXCLK: .WORD   0               ; (06) PEAK SHIFT AND CLOCK INHIBITS.
140  003314  000000         SNGLSEC: .WORD  0               ; (10) SINGLE SECTOR TO USE (0 = ALL 6).
141  003316  000000         TLMF:   .WORD   0               ; (12) G5388 TLM FLAG (1 = INSTALLED).
142  003320  160010         TCS0:   .WORD   160010          ; (14) G5388 TLM CSR ADDRESS.
143  003322                             ENDSW
(3)  003322                                                        L10003:
```

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

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

```
189          000020                  DAHS=   BIT4            ;HEAD SELECT IN SEEK
```

```
                        ;
                        ; A COUPLE OF MACROS TO REDEFINE THE ERROR CALLING CONVENTIONS
                        ; SO THAT SEQUENTIAL ERRORS ARE ASSIGNED AT ASSEMBLY TIME,
                        ; AND I DON'T HAVE TO SCREW AROUND WITH THEM !!
                        ;
000000                  SFN=    0               ; SYSTEM FATAL ERRORS RUN FROM 1 TO 99.
000143                  DFN=    99.             ; DEVICE FATAL ERRORS RUN FROM 100. UP.

                        .MACRO  SFERR   ADDR,PNTR
                        SFN=SFN+1
                        ERRSF   SFN,ADDR,PNTR
                        .ENDM

                        .MACRO  DFERR   ADDR,PNTR
                        DFN=DFN+1
                        ERRDF   DFN,ADDR,PNTR
                        .ENDM
                        ;
                        ; ANOTHER TO START EACH TEST.
                        ;
                        .MACRO  BEGIN.TEST
                                BGNTST
                                .ENDM
                        ;
                        ; AND A COUPLE OF HANDY SKIPS.
                        ;
000401                  SKP1=   BR+1
000402                  SKP2=   BR+2
000403                  SKP3=   BR+3
```

```
                        .SBTTL  GLOBAL DATA

003322  000000  UNITST: .WORD   0       ; CURRENT UNIT UNDER TEST.
003324  000000  RLCS:   .WORD   0       ; REGISTER ADDRESSES.
003326  000000  RLBA:   .WORD   0
003330  000000  RLDA:   .WORD   0
003332  000000  RLMP:   .WORD   0
003334  000000  RLBAE:  .WORD   0
003336  000000  BCSR:   .WORD   0       ; BASE ADDRESS
003340  000000  BPRIOR: .WORD   0
003342  000000  BVEC:   .WORD   0
003344  000000  TCSR:   .WORD   0       ; G5388 TLM CSR ADDRESS.
003346  000000  DRIVE:  .WORD   0
003350  000000  B.CS:   .WORD   0       ; REGISTERS BEFORE FUNCTION.
003352  000000  B.BA:   .WORD   0
003354  000000  B.DA:   .WORD   0
003356  000000  B.MP:   .WORD   0
003360  000000  B.BAE:  .WORD   0
003362  000000  DERFLG: .WORD   0
003364  000000  E.CS:   .WORD   0       ; REGISTERS AFTER FUNCTION.
003366  000000  E.BA:   .WORD   0
003370  000000  E.DA:   .WORD   0
003372  000000  E.MP:   .WORD   0
003374  000000  E.MP1:  .WORD   0       ; MAINT MODE CRC CHECK ONLY.
003376  000000  E.BAE:  .WORD   0
003400  000000  CPUTYP: .WORD   0       ; 0 = 11/03, +NZ = 11/23, -1 = UNKNOWN.P
003402  000000  RLTYP:  .WORD   0       ; RL TYPE -- 0 = RL11, 1 = RLV11, 2(3) = RLV12.
003404  120001  XPOLY:  .WORD   120001
003406  000000  BCCFBK: .WORD   0               ;LOCATION USED BY "SIMBCC"
003410  000000  CALBCC: .WORD   0               ;LOCATION USED BY "SIMBCC"
003412  000000  TEMP2:  .WORD   0               ;LOCATION USED BY "SIMBCC"
003414  000000  TEMP3:  .WORD   0               ;LOCATION USED BY "SIMBCC"
003416  000000  TEMP4:  .WORD   0               ;LOCATION USED BY "SIMBCC"
003420  000000  TEMP5:  .WORD   0               ;       DITTO
003422  000000  TEMP1:  .WORD   0               ;       DITTO
003424  000000  TMP0:   .WORD   0
003426  000000  TMP1:   .WORD   0
003430  000000  TMP2:   .WORD   0
003432  000000  GDDAT:  .WORD   0
003434  000000  BDDAT:  .WORD   0
003436  000000  INIMP:  .WORD   0               ; HOLDS INITIAL MP WORD COUNT.
003440  000000  INIDA:  .WORD   0               ;HOLDS CURRENT DA TEST WORD.
003442  000000  GDCRC3: .WORD   0               ; HOLDS CRC OF DA+3
003444  000000  GDCRC4: .WORD   0               ; HOLDA CRC OF CRC OF DA+4
003446  000233  OPIMN:  .WORD   155.
003450  001212  OPIMX:  .WORD   650.
003452  176543  HINUM:  .WORD   176543
003454  123456  LONUM:  .WORD   123456
003456  000000  TEMLO:  .WORD   0
003460  000000  TEMHI:  .WORD   0
003462  000000  DLYCNT: .WORD   0               ; 1 MSEC DELAY COUNTER.
003464  000000  DROPPED: .WORD  0               ; UNITS DROPPED COUNTER.
003466  000000  ERPOINT: .WORD  0               ; POINTS TO ONE OF THE FOLLOWING:
003470  000100  ERCOUNT: .BLKW  64.            ; ERROR COUNTERS (ENOUGH FOR 64 UNITS).
                ;
                ;PATTERNS USED FOR LOADING/READING REGISTERS
```

```
                        ;
                BEGPAT: 000000          ;GROWING 1
003670  000000          000001
003672  000001          000003
003674  000003          000007
003676  000007          000017
003700  000017          000037
003702  000037          000077
003704  000077          000177
003706  000177          000377
003710  000377          000777
003712  000777          001777
003714  001777          003777
003716  003777          007777
003720  007777          017777
003722  017777          037777
003724  037777          077777
003726  077777          177777
003730  177777          177776          ;GROWING 0
003732  177776          177774
003734  177774          177770
003736  177770          177760
003740  177760          177740
003742  177740          177700
003744  177700          177600
003746  177600          177400
003750  177400          177000
003752  177000          176000
003754  176000          174000
003756  174000          170000
003760  170000          160000
003762  160000          140000
003764  140000          100000
003766  100000          000000
003770  000000          000001          ;WALKING 1
003772  000001          000002
003774  000002          000004
003776  000004          000010
004000  000010          000020
004002  000020          000040
004004  000040          000100
004006  000100          000200
004010  000200          000400
004012  000400          001000
004014  001000          002000
004016  002000          004000
004020  004000          010000
004022  010000          020000
004024  020000          040000
004026  040000          100000
004030  100000          177777          ;WALKING 0
004032  177777          177776
004034  177776          177775
004036  177775          177773
004040  177773          177767
004042  177767          177757
004044  177757
```

```
                                        177737
                                        177677
                                        177577
                                        177377
                                        176777
                                        175777
                                        173777
                                        167777
                                        157777
                                        137777
                                        077777
                                        177777
ENDPAT:                                 000000
;
; PATTERNS FOR TESTING THE READ/WRITE BITS <9:1> OF THE CSR.
;
CSPAT:  .WORD   0                       ;SHIFTING 1
        .WORD   BIT1
        .WORD   BIT2
        .WORD   BIT3
        .WORD   BIT4
        .WORD   BIT5
        .WORD   BIT6
        .WORD   BIT8
        .WORD   BIT9
        .WORD   1576                    ;GROWING 0
        .WORD   1574
        .WORD   1570
        .WORD   1560
        .WORD   1540
        .WORD   1500
        .WORD   1400
        .WORD   1576                    ;SHIFT 0
        .WORD   1574
        .WORD   1566
        .WORD   1556
        .WORD   1536
        .WORD   1436
        .WORD   1136
        .WORD   76
        .WORD   6                       ;GROWING 1
        .WORD   16
        .WORD   36
        .WORD   76
        .WORD   176                     ;
        .WORD   576
        .WORD   1576
CSEND:  .WORD   0
;
; TABLE OF DA TEST WORDS FOR MAINTENANCE MODE CRC CALCULATIONS.
;
PATCRC: 155552
        133330
        066663
        125247
        052522
```

```
                                                                E 3
  389  004212  177774                        177774
  390  004214  000374                        000374
  391  004216  022217                        022217
  392  004220  044441                        044441
  393  004222  166663                        166663
  394  004224  144441                        144441
  395  004226  033330                        033330
  396  004230  011106                        011106
  397  004232  070704                        070704
  398  004234  107065                        107065
  399  004236  111106                        111106
  400  004240  167353                        167353
  401  004242  156732                        156732
  402  004244  146311                        146311
  403  004246  135670                        135670
  404  004250  114626                        114626
  405  004252  104205                        104205
  406  004254  073564                        073564
  407  004256  063143                        063143
  408  004260  042101                        042101
  409  004262  031460                        031460
  410  004264  021037                        021037
  411  004266  010416                        010416
  412  004270  000000            CRCEND:     000000          ; 28 TEST WORDS IN THE TABLE.
  413                                        ;
  414                                        ; TABLE OF DATA PATTERNS FOR MAINTENANCE MODE FIFO TESTS.
  415                                        ;
  416  004272  155555            PATDAT:     155555
  417  004274  133333                        133333
  418  004276  066666                        066666
  419  004300  125252                        125252
  420  004302  052525                        052525
  421  004304  177777                        177777
  422  004306  000000                        000000
  423  004310  107070                        107070
  424  004312  070707                        070707
  425  004314  144444                        144444
  426  004316  033333                        033333
  427  004320  011111                        011111
  428  004322  022222                        022222
  429  004324  044444                        044444
  430  004326  111111                        111111
  431  004330  166666                        166666
  432  004332  010421                        010421
  433  004334  021042                        021042
  434  004336  031463                        031463
  435  004340  042104                        042104
  436  004342  063146                        063146
  437  004344  073567                        073567
  438  004346  104210                        104210
  439  004350  114631                        114631
  440  004352  135673                        135673
  441  004354  146314                        146314
  442  004356  156735                        156735
  443  004360  167356                        167356
  444  004362  000000            ENDDAT:     000000          ; 28 OF THESE TOO.
```

```
                                ;
                                ; BUFFERS FOR RLV11 MAINTENANCE FUNCTION
                                ;
004364  000400          BUF1:   .BLKW   256.            ;MAINTENANCE MODE, DMA BUF1 => FIFO...
005364  000400          BUF2:   .BLKW   256.            ;...AND FIFO => BUF2.
006364  000000          BUFEND: 0
```

```
                        .SBTTL  INITIALIZATION CODE
                        ;
                        ; LOAD DEVICE PROTECTION NOT REQUIRED.
                        ;
006366                          BGNPROT
006366                                                          L$PROT::
006366 177777                   .WORD   -1              ;CSR OFFSET MAKE NOP
006370 177777                   .WORD   -1              ;MASS BUS OFFSET MAKE NOP
006372 177777                   .WORD   -1              ;DRIVE OFFSET MAKE NOP
006374                          ENDPROT
                        ;
                        ; ENTER HERE FROM DRS ON START, RESTART, CONTINUE, ETC...
                        ;
006374                          BGNINIT
006374                                                          L$INIT::
006374 012700 000034            READEF  #EF.PWR         ; POWER UP ??
006400 104447                                                           MOV     #EF.PWR,R0
                                                                        TRAP    C$REFG
006402                          BCOMPLETE RESTART
006402 103423                                                           BCS     RESTART
006404                          READEF  #EF.START       ; START ??
006404 012700 000040                                                    MOV     #EF.START,R0
006410 104447                                                           TRAP    C$REFG
006412                          BCOMPLETE START
006412 103417                                                           BCS     START
006414                          READEF  #EF.RESTART     ; RESTART ??
006414 012700 000037                                                    MOV     #EF.RESTART,R0
006420 104447                                                           TRAP    C$REFG
006422                          BCOMPLETE RESTART
006422 103413                                                           BCS     RESTART
006424                          READEF  #EF.NEW         ; NEW PASS ??
006424 012700 000035                                                    MOV     #EF.NEW,R0
006430 104447                                                           TRAP    C$REFG
006432                          BCOMPLETE NEWPAS
006432 103472                                                           BCS     NEWPAS
006434                          READEF #EF.CONTINUE     ; CONTINUE ??
006434 012700 000036                                                    MOV     #EF.CONTINUE,R0
006440 104447                                                           TRAP    C$REFG
006442                          BCOMPLETE 1$
006442 103401                                                           BCS     1$
006444 000506                   BR      NXTU            ; NONE OF THE ABOVE.
006446 000137 007050    1$:     JMP     CONT
                        ;
006452                  START:RESTART:
006452 000005                   RESET
006454 005737 003306            TST     PDSW            ; PROM DUMP ??
006460 001402                   BEQ     1$              ; BR IF NOT.
006462 000137 024606            JMP     ROMDUMP         ; *** PROM DUMPER, NO RETURN ***
006466 012737 176543 003452 1$: MOV     #176543,HINUM   ;RANDOM GEN. PRIMES.
006474 012737 123456 003454     MOV     #123456,LONUM
006502 005037 003464            CLR     DROPPED         ; CLEAR UNITS DROPPED COUNTER.
006506 012701 003470            MOV     #ERCOUNT,R1     ; GET ERROR COUNTER POINTER.
006512 012700 000100            MOV     #64.,R0
006516 005021           2$:     CLR     (R1)+           ; CLEAR ALL ERROR COUNTERS.
006520 005300                   DEC     R0
006522 001375                   BNE     2$
```

```
                                                                    H 3
 491  006524  013746  000004          ENVIRN: MOV     @#4,-(SP)       ; SAVE VECTOR 4...
 492  006530  013746  000010                  MOV     @#10,-(SP)      ;...AND 10...
 493  006534  010605                         MOV     SP,R5           ;...AND THE STACK POINTER.
 494  006536  012737  006574  000004          MOV     #LSI,@#4        ; SET BUS TRAP.
 495  006544  005000                         CLR     R0
 496  006546  005737  177776                 TST     @#177776        ; PSW ADDRESS VALID ??
 497  006552  000240                         NOP                     ; TRAP THRU 4 IF NOT (LSI11 OR 11/2).
 498  006554  012737  006570  000010          MOV     #NLSI,@#10      ; SET ILLEGAL TRAP.
 499          000007                         MFPT=   7
 500  006562  000007                         MFPT                    ;GET PROCESSOR TYPE (IF POSSIBLE).
 501  006564  000240                         NOP                     ; TRAP THRU 10 IF NOT (NOT 11/23).
 502  006566  000402                         SKP2
 503  006570  012700  177777          NLSI:  MOV     #-1,R0          ; CPU TYPE UNKNOWN.
 504  006574  010037  003400          LSI:   MOV     R0,CPUTYP       ; CPU IS LSI SOMETHING 'R OTHER.
 505  006600  010506                         MOV     R5,SP           ; RESTORE STACK...
 506  006602  012637  000010                 MOV     (SP)+,@#10      ;...AND VECTORS.
 507  006606  012637  000004                 MOV     (SP)+,@#4
 508  006612  004737  011242                 JSR     PC,..SIZE       ; SIZE AVAILABLE MEMORY.
 509  006616                                 DORPT                   ; REPORT OPERATING ENVIRONMENT.
 (3)  006616  104424                                                                    TRAP    C$DRPT
 510
 511  006620  023737  003464  002012  NEWPAS: CMP     DROPPED,L$UNIT  ; UNITS STILL ALIVE ??
 512  006626  002401                         BLT     1$
 513  006630                                 DOCLN                   ; NO, ABORT
 (3)  006630  104444                                                                    TRAP    C$DCLN
 514  006632  012737  003466  003466  1$:    MOV     #ERCOUNT-2,ERPOINT ;INIT THE UNIT ERROR POINTER.
 515  006640  013737  003454  003456          MOV     LONUM,TEMLO
 516  006646  013737  003452  003460          MOV     HINUM,TEMHI     ; NEW PRIMES FOR NEW PASS.
 517  006654  012737  177777  003322          MOV     #-1,UNITST      ; RESET UNIT NUMBER.
 518  006662  005237  003322          NXTU:  INC     UNITST          ; BUMP UNIT NUMBER...
 519  006666  062737  000002  003466          ADD     #2,ERPOINT      ;...AND ERROR COUNT POINTER.
 520  006674  023737  003322  002012          CMP     UNITST,L$UNIT
 521  006702  002346                         BGE     NEWPAS          ;
 522  006704                          1$:    GPHARD  UNITST,R0
 (3)  006704  013700  003322                                                           MOV     UNITST,R0
 (3)  006710  104442                                                                    TRAP    C$GPHRD
 523  006712                                 BNCOMPLETE NXTU         ; BR IF UNIT NOT THERE (DROPPED).
 (2)  006712  103363                                                                    BCC     NXTU
 524  006714  012037  003336          2$:    MOV     (R0)+,BCSR      ; SET UP RUN TIME P-TABLE...
 525  006720  005737  036704                    TST     MIMIC           ; ****** IF MIMIC
 526  006724  001403                          BEQ     3$              ; ******
 527  006726  012737  037224  003336          MOV     #DUMMY,BCSR     ; ****** USE DUMMY REGISTERS
 528  006734  012037  003342          3$:    MOV     (R0)+,BVEC      ;...FOR THIS UNIT.
 529  006740  012037  003340                  MOV     (R0)+,BPRIOR
 530  006744  012037  003346                  MOV     (R0)+,DRIVE
 531  006750  012037  003402                  MOV     (R0)+,RLTYP     ; SET CONTROLLER TYPE.
 532
 533  006754  013737  003316  003344          MOV     TLMF,TCSR       ; RUNNING WITH TLM'S ??
 534  006762  001412                         BEQ     5$              ; BR IF NOT.
 535  006764  013737  003320  003344          MOV     TCS0,TCSR       ; YES, GET BASE TLM CSR...
 536  006772  013700  003322                  MOV     UNITST,R0       ;...AND CURRENT UNIT NUMBER.
 537  006776  001404                         BEQ     5$              ; BR IF UNIT IS ZERO.
 538  007000  062737  000010  003344  4$:    ADD     #10,TCSR        ; OTHERWISE, ADJUST FOR CURRENT UNIT.
 539  007006  077004                         SOB     R0,4$
 540
 541  007010  005737  003400          5$:    TST     CPUTYP
```

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

```
                                                                                                    L10006:
(3)  007270                                                                                                 TRAP    C$AUTO
(3)  007270  104461
589
590                                 ;
591                                 ; CLEAN-UP CODING SECTION (DO AT END-PASS (OR SUB PASS)).
592                                 ;
592  007272                                 BGNCLN
(3)  007272                                                                                         L$CLEAN::
593  007272  012700  000012                MOV     #10.,R0
594  007276  032777  000200  174020  1$:   BIT     #CRDY,@RLCS     ; GIVE HIM ABOUT 100 MSEC...
595  007304  001004                        BNE     2$              ;...TO FINISH UP.
596  007306  004537  010160                JSR     R5,WDELAY
597  007312  000012                        10.
598  007314  077010                        SOB     R0,1$
599  007316  012777  000200  174000  2$:   MOV     #CRDY,@RLCS     ; THEN CLEAR ALL OPTION BITS.
600  007324  023727  003402  000003        CMP RLTYP,#RLV12X       ; BAE IN USE ??
601  007332  001002                        BNE     3$
602  007334  005077  173774                CLR     @RLBAE          ; IF SO, MAKE SURE IT'S CLEAR.
603  007340                        3$:     SETPRI  #PRI07
(3)  007340  012700  000340                                                                                 MOV     #PRI07,R0
(3)  007344  104441                                                                                         TRAP    C$SPRI
604  007346                                 ENDCLN
(3)  007346                                                                                         L10007:
(3)  007346  104412                                                                                         TRAP    C$CLEAN
605
606                                 ;
607                                 ; DROP UNITS -- AND KEEP TRACK OF 'EM.
608                                 ;
608  007350                                 BGNDU
(3)  007350                                                                                         L$DU::
609  007350  023737  002012  003464        CMP     L$UNIT,DROPPED  ; ALL UNITS DROPPED ??
610  007356  001424                        BEQ     2$              ; YUP, DO NOTHING.
611  007360  005237  003464                INC     DROPPED         ; NOT YET, BUMP COUNT...
612  007364                                 PRINTF  #1$             ;...AND TELL THE MAN.
(7)  007364  012746  007406                                                                                 MOV     #1$,-(SP)
(6)  007370  012746  000001                                                                                 MOV     #1,-(SP)
(3)  007374  010600                                                                                         MOV     SP,R0
(4)  007376  104417                                                                                         TRAP    C$PNTF
(4)  007400  062706  000004                                                                                 ADD     #4,SP
613  007404  000411                        BR      2$
614  007406  040445  047125  052111  1$:   .ASCIZ  /%AUNIT DROPPED%N/
615  007430                                 .EVEN
616  007430                        2$:     ENDDU
(3)  007430                                                                                         L10010:
(3)  007430  104453                                                                                         TRAP    C$DU
617
618                                 ;
619                                 ; IF THEY'RE PICKED UP AGAIN, TWEAK THE COUNTER.
620                                 ;
620  007432                                 BGNAU
(3)  007432                                                                                         L$AU::
621  007432  005737  003464                TST     DROPPED         ; ALL UNITS BACK IN PLACE ??
622  007436  001423                        BEQ     2$              ; JUST EXIT IF SO.
623  007440  005337  003464                DEC     DROPPED         ; NOT YET, ADJUST COUNT...
624  007444                                 PRINTF  #1$             ;...AND TELL HIM.
(7)  007444  012746  007466                                                                                 MOV     #1$,-(SP)
(6)  007450  012746  000001                                                                                 MOV     #1,-(SP)
(3)  007454  010600                                                                                         MOV     SP,R0
(4)  007456  104417                                                                                         TRAP    C$PNTF
```

```
                                                                                    K 3
    (4)  007460  062706  000004                                                        ADD     #4,SP
625  007464  000410                    BR      2$
626  007466  040445  047125  052111  1$:     .ASCIZ  /%AUNIT ADDED%N/
627          007506                    .EVEN
628  007506                         2$:     ENDAU
    (3)  007506                                                                 L10011:
    (3)  007506  104452                                                                TRAP    C$AU
```

```
                        .SBTTL  GLOBAL SUBROUTINES
                        ;
                        ;SUBROUTINE TO CHECK FOR CONTROLLER ERRORS.
                        ;TRANSLATE THE ERROR BITS INTO PLAIN LANGUAGE AND MERGE THEM
                        ;INTO THE ERROR BUFFER (EM99) FOR SUBSEQUENT PRINTING.
                        ;DRIVE ERROR IS INCLUDED, BUT IS IGNORED IN THE DISKLESS TESTS.
                        ;
                        ;CALL:  JSR     R5,GETERR
                        ;       BR      XX      ; NO ERRORS FOUND, RETURN.
                        ;       ERRXX           ; ERROR, RETURN.
                        ;
007510 010146           GETERR: MOV     R1,-(SP)        ; SAVE R1.
007512 005037 003362            CLR     DERFLG          ;CLEAR OUT DRIVE ERROR FLAG
007516 012701 034404            MOV     #EM99,R1        ;SET TEXT BUFFER POINTER.
007522 032737 176000 003364     BIT     #176000,E.CS    ; ANY ERRORS THERE ??
007530 001004                   BNE     1$              ; BR IF SO.
007532 004537 010004            JSR     R5,FIX          ; NO, JUST SAY "NONE"...
007536 030630                   NON
007540 000516                   BR      10$             ;...AND RETURN.
                        ;
007542 005737 003364    1$:     TST     E.CS            ;IS COMPOSITE ERROR SET ? (BETTER BE)
007546 100003                   BPL     2$              ;IF NOT SOMETHING'S WRONG !!!
007550 004537 010004            JSR     R5,FIX          ;YES, PUT "CERR" IN STRING.
007554 030551                   CERR
007556 032737 040000 003364 2$: BIT     #DERR,E.CS      ;DRIVE ERROR SET?
007564 001411                   BEQ     3$              ; BR IF NOT.
007566 005237 003362            INC     DERFLG          ;YES, SET DRV ERROR FLAG
007572 004537 010004            JSR     R5,FIX          ; ADD "DRV" TO STRING.
007576 030556                   DEMES
007600 032737 036000 003364     BIT     #036000,E.CS    ; ANY OTHERS ???
007606 001473                   BEQ     10$             ; NO, IGNORE THE DRIVE ERROR (DISKLESS).
007610 005725           3$:     TST     (R5)+           ; YES, BUMP PC TO TAKE ERROR RETURN.
007612 032737 020000 003364     BIT     #NXM!PAR,E.CS   ; NON-EX OR PARITY ??
007620 001415                   BEQ     5$              ; BR IF NOT.
007622 012737 030563 007652     MOV     #NXMMES,4$
007630 032737 002000 003364     BIT     #OPI,E.CS
007636 001403                   BEQ     .+10
007640 012737 030570 007652     MOV     #PARMES,4$
007646 004537 010004            JSR     R5,FIX          ; ADD "NXM" OR "PAR".
007652 030563           4$:     NXMMES
007654 032737 010000 003364 5$: BIT     #DLT!HNF,E.CS   ; DATA LATE OR HEADER NOT FOUND ??
007662 001415                   BEQ     7$              ; BR IF NOT.
007664 012737 030575 007714     MOV     #DLTMES,6$
007672 032737 002000 003364     BIT     #OPI,E.CS
007700 001403                   BEQ     .+10
007702 012737 030602 007714     MOV     #HNFMES,6$
007710 004537 010004            JSR     R5,FIX          ; ADD "DLT" OR "HNF".
007714 030575           6$:     DLTMES
007716 032737 004000 003364 7$: BIT     #DCRC!HCRC,E.CS ; EITHER CRC ??
007724 001415                   BEQ     9$              ; BR IF NOT.
007726 012737 030607 007756     MOV     #DCRCMES,8$
007734 032737 002000 003364     BIT     #OPI,E.CS
007742 001403                   BEQ     .+10
007744 012737 030615 007756     MOV     #HCRCMES,8$
007752 004537 010004            JSR     R5,FIX          ; ADD "DCRC" OR "HCRC".
007756 030607           8$:     DCRCMES
```

```
                                        9$:     BIT     #OPI,E.CS       ; "OPI" SET ??
                                                BEQ     10$
                                                JSR     R5,FIX          ; ADD "OPI" TO STRING.
                                                OPIMES
                                        10$:    CLRB    (R1)            ; TERMINATE ERROR LIST...
                                                MOV     (SP)+,R1        ;...AND RESTORE R1.
                                                RTS     R5              ; RETURN.
                                        ;
                                        ;ROUTINE TO MOVE ASCII STRINGS
                                        ;USES REGISTERS R1 - WHERE STRING IS BEING BUILT
                                        ;
                                        ;       CALL    JSR     R5,FIX
                                        ;               .WORD           ;ADDRESS OF STRING TO MOVE
                                        ;
                                        FIX:    MOV     (R5)+,R0        ;GET ADDRESS AND MOVE RETURN
                                        1$:     MOVB    (R0)+,(R1)+     ;GET BYTE AND UPDATE
                                                BNE     1$              ;WATCH 0 BYTE TERMINATOR
                                                TSTB    -(R1)           ;BACK UP OVER ZERO BYTE
                                                RTS     R5              ;EXIT
                                        ;
                                        ;LOAD REGISTERS BEFORE FUNCTION
                                        ;CALL:  JSR     R5,BEFORE
                                        ;
010016  017737  173302  003350          BEFORE: MOV     @RLCS,B.CS      ;READ CS...
010024  042737  176000  003350                  BIC     #176000,B.CS    ;...ERRORS WILL BE CLEARED ON XCT.
010032  017737  173270  003352                  MOV     @RLBA,B.BA      ;READ BA
010040  017737  173264  003354                  MOV     @RLDA,B.DA      ;READ DA
010046  013737  003436  003356                  MOV     INIMP,B.MP      ; INITIAL MP IS THE WORD COUNT.
010054  023727  003402  000003                  CMP     RLTYP,#RLV12X
010062  001003                                  BNE     1$
010064  017737  173244  003360                  MOV     @RLBAE,B.BAE    ; READ BAE IF ENABLED.
010072  000205                          1$:     RTS     R5
                                        ;
                                        ;LOAD REGISTERS AT ERROR OR DONE.
                                        ;CALL:  JSR     R5,AFTER
                                        ;
010074  017737  173224  003364          AFTER:  MOV     @RLCS,E.CS      ;READ CS
010102  017737  173220  003366                  MOV     @RLBA,E.BA      ;READ BA
010110  017737  173214  003370                  MOV     @RLDA,E.DA      ;READ DA
010116  017737  173210  003372                  MOV     @RLMP,E.MP      ;READ MP
010124  005737  036704                            TST   MIMIC           ; ****** IF MIMIC
010130  001003                                    BNE   1$              ; ****** BYPASS 2ND MP READ
010132  017737  173174  003374                  MOV     @RLMP,E.MP1     ;READ MP (MAINT 2ND CRC CHECK ONLY).
010140  023727  003402  000003          1$:     CMP     RLTYP,#RLV12X
010146  001003                                  BNE     2$
010150  017737  173160  003376                  MOV     @RLBAE,E.BAE    ; READ BAE IF ENABLED.
010156  000205                          2$:     RTS     R5
                                        ;
                                        ; ROUTINE TO DELAY IN 1 MSEC INCREMENTS.
                                        ; CALIBRATED FOR LSI'S ONLY.
                                        ; DELAY COUNT IS 291. (443B) FOR LSI 11/23.
                                        ;                120. (170B) FOR LSI 11 OR 11/2.
                                        ;                500. (764B) IF CPU TYPE IS UNKNOWN.
                                        ;
                                        ; CALL: JSR R5,WDELAY
                                        ;       N                       ; DELAY N MSECS.
```

```
                                        ;
010160  010146                  WDELAY: MOV     R1,-(SP)
010162  010246                          MOV     R2,-(SP)
010164  012502                          MOV     (R5)+,R2        ;APPROX MSEC DELAY
010166  013701  003462          2$:     MOV     DLYCNT,R1       ; GET 1 MSEC TIMER.
010172  005301                  3$:     DEC     R1              ;START LOOP
010174  001376                          BNE     3$
010176  005302                          DEC     R2              ;CHECK ON MSECS REQUESTED
010200  001372                          BNE     2$              ;BRANCH AND DO ANOTHER LOOP
010202  012602                  4$:     MOV     (SP)+,R2        ;SETUP FOR RETURN AFTER DELAY
010204  012601                          MOV     (SP)+,R1
010206  000205                          RTS     R5
                                        ;
                                        ;ROUTINE TO LOAD RLCS WITH RLV11/12 MAINTENANCE FUNCTION
                                        ;EITHER FLAG DRIVEN OR INTERRUPT MODE.
                                        ;
                                        ;CALL:  JSR     R5,LDFUN
                                        ;       .WORD   MAINT   ; OR MAINT!INTEN
                                        ;       .WORD   -N      ; WORD COUNT.
                                        ;
                                        ; LOCATIONS "BA16" AND "BA22" COMPRISE A 22 BIT PHYSICAL BUS ADDRESS
                                        ; AND MUST BE SET BY THE CALLING ROUTINE PRIOR TO ENTRY.
                                        ;
010210  012500                  LDFUN:  MOV     (R5)+,R0        ;BUILD THE FUNCTION IN R0.
010212  012537  003436                  MOV     (R5)+,INIMP     ; SAVE WORD COUNT FOR LATER.
010216  042700  177661                  BIC     #177661,R0      ;MASK GARBAGE BITS.
010222  053700  003346                  BIS     DRIVE,R0        ;INSERT DRIVE (0)...
010226  052700  000200                  BIS     #CRDY,R0        ;...AND THE READY BITS.
010232  010077  173066                  MOV     R0,@RLCS        ; LOAD UP THE CSR.
010236  013777  010352  173062          MOV     BA16,@RLBA      ; LOAD BUS ADDRESS <15:0>.
010244  023727  003402  000003          CMP     RLTYP,#RLV12X   ; RLV12 WITH BAE ??
010252  001413                          BEQ     1$              ; BR IF SO.
010254  013700  010354                  MOV     BA22,R0         ; NO, GET EXTENSION IF ANY...
010260  042700  177774                  BIC     #^C3,R0         ;...KEEP <17:16> ONLY...
010264  006300                          ASL     R0
010266  006300                            ASL   R0
010270  006300                              ASL R0
010272  006300                                ASL R0            ;...MOVE 'EM TO <5:4>...
010274  050077  173024                  BIS     R0,@RLCS        ;...AND INSERT INTO CS.
010300  000403                          BR      2$
010302  013777  010354  173024  1$:     MOV     BA22,@RLBAE     ; RLV12X, SET BA EXTENSION <21:16>.
010310  013777  003440  173012  2$:     MOV     INIDA,@RLDA     ;LOAD DISK ADDRESS.
010316  013777  003436  173006          MOV     INIMP,@RLMP     ;LOAD WORD COUNT.
010324  004537  010016                  JSR     R5,BEFORE       ;SAVE STATE BEFORE FUNCTION.
010330  042777  000200  172766          BIC     #CRDY,@RLCS     ;XCT MAINT FUNCTION.
010336  005737  036704                  TST     MIMIC           ;****** IF MIMIC
010342  001402                          BEQ     3$              ;******
010344  004737  036706                  JSR     PC,EMURLV       ;****** EMULATE THE RLV FUNCTION
010350  000205                  3$:     RTS     R5              ;RETURN
                                        ;
010352  004364                  BA16:   BUF1            ; 16 BIT BUFFER ADDRESS <15:0>,
010354  000000                  BA22:   0               ; 6 BIT EXTENSION <21:16> IN <5:0>.
                                        ;
                                        ;ROUTINE TO FILL BUFFER WITH DATA PATTERNS FOR RLV MAINTENANCE.
                                        ;BUF1 IS FILLED WITH 256 WORD PATTERN (OR COMPLIMENTING PATTERN).
                                        ;BUF2 IS FILLED WITH 255 ZEROS AND 123456.
```

```
                                        ;
                                        ;CALL:  JSR     R5,SETPAT(SETCMP)
                                        ;       .WORD   N       ;PATTERN FOR BUFFER
                                        ;
010356  005000                          SETPAT: CLR     R0              ; SET NO COMPLIMENT...
010360  000402                                  BR      .+6             ;...AND SKIP NEXT.
010362  012700  177777                  SETCMP: MOV     #-1,R0          ; SET COMPLIMENT.
010366  010146                                  MOV     R1,-(SP)
010370  010246                                  MOV     R2,-(SP)
010372  012537  003424                          MOV     (R5)+,TMP0      ; DATA TO STORE.
010376  012701  004364                          MOV     #BUF1,R1        ;BUFFER POINTER.
010402  012702  000400                          MOV     #256.,R2        ;WORD COUNT
010406  013721  003424                  1$:     MOV     TMP0,(R1)+
010412  005700                                  TST     R0              ; ARE WE COMPLIMENTING ??
010414  001402                                  BEQ     2$              ; NO, SKIP NEXT.
010416  005137  003424                          COM     TMP0            ; YES, COMPLIMENT IT.
010422  005302                          2$:     DEC     R2
010424  001370                                  BNE     1$
010426  012702  000377                          MOV     #255.,R2        ; NOW CLEAR THE REST OF...
010432  005021                          3$:     CLR     (R1)+           ;...THE BUFFER.
010434  005302                                  DEC     R2
010436  001375                                  BNE     3$
010440  012721  123456                          MOV     #123456,(R1)+   ;STORE IN LAST BUFFER WORD.
010444  012602                                  MOV     (SP)+,R2
010446  012601                                  MOV     (SP)+,R1
010450  000205                                  RTS     R5
                                        ;
                                        ;ROUTINE TO SETUP BUFFER WITH RANDOM NUMBERS FOR RLV11 MAINT. FUNCTION
                                        ;SAME PATTERN IS USED FOR EACH CONTROLLER
                                        ;END OF PASS WILL CHANGE RANDOM PATTERN PRIMES
                                        ;       CALL    JSR     R5,SETRAN
                                        ;
010452  010146                          SETRAN: MOV     R1,-(SP)
010454  010246                                  MOV     R2,-(SP)
010456  012701  004364                          MOV     #BUF1,R1        ;FIRST BUFFER START
010462  012702  000400                          MOV     #256.,R2        ;BUFFER COUNT
010466  004537  010526                  1$:     JSR     R5,RAND         ;GET RANDOM NUMBER
010472  013721  003452                          MOV     HINUM,(R1)+     ;STORE IN BUFFER
010476  005302                                  DEC     R2
010500  001372                                  BNE     1$
010502  012702  000377                          MOV     #255.,R2        ; NOW CLEAR THE REST.
010506  005021                          2$:     CLR     (R1)+
010510  005302                                  DEC     R2
010512  001375                                  BNE     2$
010514  012721  123456                          MOV     #123456,(R1)+   ;STORE IN LAST BUFFER WORD
010520  012602                                  MOV     (SP)+,R2
010522  012601                                  MOV     (SP)+,R1
010524  000205                                  RTS     R5
                                        ;
                                        ;THIS ROUTINE IS A DOUBLE PRECISION PSEUDO RANDOM NUMBER GENERATOR
                                        ;WITH A RANGE OF 0 TO 2(+33)-1.
                                        ;CALL:
                                        ;       JSR     R5,RAND         ;CALL THE ROUTINE
                                        ;       RETURN                  ;RETURN HERE THE RANDOM NUMBER
                                        ;                               ;WILL BE IN HINUM,LONUM
                                        ;
```

```
                                        C 4
010526  010146                  RAND:   MOV     R1,-(SP)        ;PUSH R1 ON STACK
010530  010246                          MOV     R2,-(SP)        ;PUSH R2 ON STACK
010532  010346                          MOV     R3,-(SP)        ;PUSH R3 ON STACK
010534  013703  003454                  MOV     LONUM,R3        ;SET R3 WITH LOW
010540  013701  003452                  MOV     HINUM,R1        ;SET R1 WITH HIGH
010544  012702  177771                  MOV     #-7,R2          ;SET SHIFT COUNTER
010550  006303                  1$:     ASL     R3              ;SHIFT R3 LEFT AND
010552  006101                          ROL     R1              ;ROTATE CARRY INTO R1 AND
010554  005202                          INC     R2              ;CHECK FOR DONE
010556  001374                          BNE     1$              ;CONTINUE SHIFT LOOP
010560  063703  003454                  ADD     LONUM,R3        ;ADD NUMBER TO MAKE X 129
010564  005501                          ADC     R1              ;PROPOGATE CARRY
010566  063701  003452                  ADD     HINUM,R1        ;ADD NUMBER TO MAKE X 129
010572  062703  001057                  ADD     #1057,R3        ;ADD LOW CONSTANT
010576  005501                          ADC     R1              ;PROPOGATE CARRY
010600  062701  047401                  ADD     #47401,R1       ;ADD HIGH CONSTANT
010604  010337  003454                  MOV     R3,LONUM        ;SAVE R3
010610  010137  003452                  MOV     R1,HINUM        ;SAVE R1
010614  012603                          MOV     (SP)+,R3        ;POP STACK INTO R3
010616  012602                          MOV     (SP)+,R2        ;POP STACK INTO R2
010620  012601                          MOV     (SP)+,R1        ;POP STACK INTO R1
010622  000205                          RTS     R5              ;RETURN
                                ;
                                ; ROUTINES TO SET A FLAG ON BUS-ERROR AND/OR RL INTERRUPT.
                                ;
010624  005227                  TRAP4:  INC     (PC)+
010626  000000                  TRPFLG: 0
010630  000002                          RTI
010632  005227                  RLINT:  INC     (PC)+
010634  000000                  INTFLG: 0
010636  000002                  CLKRTI: RTI                     ; USE TO DISMISS FREE-RUNNING CLOCK.
                                ;
                                ;ROUTINE TO WAIT FOR CONTROLLER READY
                                ; THE 800 MSEC INTERVAL APPLIES TO LSI'S ONLY !!!
                                ; IN ANY CASE, IT'S JUST A KEEP-ALIVE TIMER.
                                ;  ON EXIT, -4(SP) HOLDS THE REMNANTS OF THE 800. MS COUNT.
                                ;
010640  012746  001440          WTCRDY: MOV     #800.,-(SP)     ;SET 800 MSEC TIMER.
010644  032777  000200  172452  1$:     BIT     #CRDY,@RLCS     ;CONTROLLER READY ??
010652  001014                          BNE     2$              ;YES, EXIT
010654  004537  010160                  JSR     R5,WDELAY       ;WAIT A WHILE
010660  000001                          1                       ;APPROX A MILLISECOND
010662  005316                          DEC     (SP)            ;CHECK IF TIME UP
010664  001367                          BNE     1$              ;NO GO BACK
010666  004537  010074                  JSR     R5,AFTER        ; GET REGISTERS FOR ERROR.
010672                                  DFERR   CRTIM,ERRO      ;CONTROLLER TIMED OUT
(5)  010672  104455                                                             TRAP    C$ERDF
(6)  010674  000144                                                             .WORD   100
(6)  010676  030476                                                             .WORD   CRTIM
(6)  010700  026164                                                             .WORD   ERRO
010702  000402                          SKP2
010704  004537  010074          2$:     JSR     R5,AFTER        ;GET REGISTERS
010710  005726                          TST     (SP)+           ; CLEAN THE STACK...
010712  000205                          RTS     R5              ;...AND RETURN.
                                ;
```

```
                                        ;RLV11 MAINTENANCE SUBROUTINE FOR CRC CALCULATIONS
                                        ;ROUTINE TO RETRIEVE PATTERN AND CALCULATE CRC OF PATTERN+3
                                        ;AND CRC OF CRC OF PATTERN+4.
                                        ;PATTERN IS SAVED IN "INIDA" FOR SUBSEQUENT LOADING FOR DA.
                                        ;CRC OF PATTERN+3 WILL BE STORED IN "GDCRC3".
                                        ;CRC OF CRC OF PATTERN+4 WILL BE STORED IN "GDCRC4".
                                        ;
                                        ;CALL:   JSR     R5,CALCRC
                                        ;        .WORD   N       ;PATTERN FOR INITIAL DA
                                        ;
010714  012537  003440          CALCRC: MOV     (R5)+,INIDA     ;STORE PATTERN
010720  013737  003440  003422          MOV     INIDA,TEMP1
010726  113737  003422  003420          MOVB    TEMP1,TEMP5
010734  062737  000003  003420          ADD     #3.,TEMP5       ;ADD 3 TO PATTERN
010742  113737  003420  003422          MOVB    TEMP5,TEMP1
010750  013737  003422  010764          MOV     TEMP1,1$
010756  004537  011060                  JSR     R5,SIMBCC       ;CALCULATE EXPECTED CRC
010762  000020                          16.                     ;DATA BITS
010764  000000                  1$:     .WORD   0               ;INITIAL PATTERN+3
010766  000000                          .WORD   0
010770  013737  003410  003442          MOV     CALBCC,GDCRC3   ;SAVE CRC OF PATTERN+3
010776  005237  003420                  INC     TEMP5           ;VALUE=PATTERN+4
011002  113737  003420  003422          MOVB    TEMP5,TEMP1
011010  013737  003422  011024          MOV     TEMP1,2$
011016  004537  011060                  JSR     R5,SIMBCC       ;CALCULATE EXPECTED CRC
011022  000020                          16.                     ;DATA BITS
011024  000000                  2$:     .WORD   0               ;INITIAL PATTERN+4
011026  000000                          .WORD   0               ;STARTING CRC=0
011030  013737  003410  011044          MOV     CALBCC,3$       ;STORE CRC FOR NEXT CALL
011036  004537  011060                  JSR     R5,SIMBCC       ;CAL. CRC OF CRC OF DA+4
011042  000020                          16.                     ;DATA BITS
011044  000000                  3$:     .WORD   0               ;CRC OF DA+4
011046  000000                          .WORD   0               ;STARTING CRC=0
011050  013737  003410  003444          MOV     CALBCC,GDCRC4   ;SAVE CRC OF CRC OF DA+4
011056  000205                          RTS     R5
                                        ;
                                        ; SUBROUTINE TO CALCULATE A CRC.
                                        ;ROUTINE WILL CALCULATE A CRC-16 CRC ON A WORD OF
                                        ;1-16 BITS IN LENGTH, RESULT IS RETURNED IN "CALBCC"
                                        ;
                                        ;       CALL:   JSR     R5,SIMBCC
                                        ;               .WORD           ;NUMBER OF BITS (1-16)
                                        ;               .WORD           ;DATA FOR CRC CALCULATION
                                        ;               .WORD           ;PREVIOUS OR STARTING CRC
                                        ;                               (SHOULD BE ZEROED FOR START)
                                        ;
011060  010046                  SIMBCC: MOV     R0,-(SP)        ;SAVE R0
011062  010146                          MOV     R1,-(SP)        ;SAVE R1
011064  010246                          MOV     R2,-(SP)        ;SAVE R2
011066  012537  003412                  MOV     (R5)+,TEMP2     ;GET NUMBER OF BITS
011072  012537  003414                  MOV     (R5)+,TEMP3     ;GET DATA FOR CRC CALCULATION
011076  012537  003416                  MOV     (R5)+,TEMP4     ;GET STARTING CRC
011102  005037  003406          1$:     CLR     BCCFBK          ;
011106  013700  003416                  MOV     TEMP4,R0        ;GET PRESENT CRC
011112  006037  003414                  ROR     TEMP3           ;ROTATE NEW DATA
011116  005500                          ADC     R0              ;MERGE NEW WITH OLD
```

```
                                                                E 4
 962  011120  032700  000001          BIT     #1,R0           ;BIT 0 SET
 963  011124  001402                  BEQ     2$              ;IF NOT CONTINUE
 964  011126  005137  003406          COM     BCCFBK          ;
 965  011132  013700  003404  2$:     MOV     XPOLY,R0        ;GET CRC POLYNOMIAL (CRC-16)
 966  011136  005100                  COM     R0              ;COMPLIMENT POLYNOMIAL
 967  011140  040037  003406          BIC     R0,BCCFBK
 968  011144  000241                  CLC                     ;CLEAR CARRY
 969  011146  006037  003416          ROR     TEMP4
 970  011152  013700  003406          MOV     BCCFBK,R0
 971  011156  013701  003416          MOV     TEMP4,R1
 972  011162  010102                  MOV     R1,R2
 973  011164  040100                  BIC     R1,R0
 974  011166  043702  003406          BIC     BCCFBK,R2
 975  011172  050200                  BIS     R2,R0
 976  011174  043737  003404  003416  BIC     XPOLY,TEMP4
 977  011202  050037  003416          BIS     R0,TEMP4
 978  011206  005337  003412          DEC     TEMP2
 979  011212  001333                  BNE     1$
 980  011214  013737  003416  003410  MOV     TEMP4,CALBCC
 981  011222  012602                  MOV     (SP)+,R2
 982  011224  012601                  MOV     (SP)+,R1
 983  011226  012600                  MOV     (SP)+,R0
 984  011230  000205                  RTS     R5              ;RETURN
```

```
                        .SBTTL  MEMORY SIZER
                        ;*****************************************************************
                        ;
                        ; ROUTINE TO SIZE AVAILABLE MEMORY.
                        ; FREELY ADAPTED FROM ".$SIZE" IN SYSMAC.SML (C3).
                        ; THIS ROUTINE MUST RESIDE WITHIN THE FIRST 24K (0-137776).
                        ;
                        ; USE KT IF IT'S AVAILABLE, OTHERWISE SIZE UP TO 30K IN THE
                        ; TRADITIONAL FASHION.  RETURN WITH:
                        ;
011232  000000          .MSIZE: .WORD   0       ; TOTAL MEMORY SIZE (K WORDS).
011234  000000          .LSTPG: .WORD   0       ; PAGE ADDRESS (PAF) OF LAST 1K PAGE...
                                                ;...OR ZERO IF KT NOT AVAILABLE.
011236  000000          .LSTAD: .WORD   0       ; LAST VIRTUAL ADDRESS IN LAST PAGE...
                                                ;...OR LAST ADDRESS UNDER 30K (IF NO KT).
011240  000020          .ABUSW: .WORD   16.     ; ADDRESS BUS WIDTH, 16, 18, OR 22.
                        ;
                        ; ALL GENERAL REGISTERS ARE USED BUT NOT SAVED.
                        ; MEMORY PARITY ERRORS (IF ANY) ARE IGNORED.
                        ;
                177572  MMR0=   177572          ; KT CONTROL REGISTERS.
                177574  MMR1=   177574
                177576  MMR2=   177576
                172516  MMR3=   172516
                172340  KIPAR0= 172340          ; KERNAL, I SPACE, PAR 0.
                172300  KIPDR0= KIPAR0-40       ;                  PDR 0.
                        ;
                        ;*****************************************************************
011242  012737  000020  011240  .SIZE:  MOV     #16.,.ABUSW     ; ASSUME 16 BIT ADDRESSING.
011250  013746  000004          MOV     @#4,-(SP)       ; SAVE BUS-ERROR...
011254  013746  000006          MOV     @#6,-(SP)
011260  013746  000114          MOV     @#114,-(SP)     ;...AND PARITY VECTORS.
011264  013746  000116          MOV     @#116,-(SP)
011270  010605                  MOV     SP,R5           ; SAVE STACK POINTER IN R5.
                        
011272  012737  000114  000114          MOV     #116,@#114      ; IGNORE PARITY ERRORS.
011300  012737  000002  000116          MOV     #RTI,@#116
011306  012737  011335  000004          MOV     #1$,@#4         ; SIZE USING "1$" IF KT ISN'T THERE.
011314  012737  000340  000006          MOV     #340,@#6
011322  005737  177572                  TST     MMR0
011326  000240                          240
011330  000417                  BR      .SIZKT          ; OTHERWISE, SIZE USING THE KT.
                        
011332  012737  011360  000004  1$:     MOV     #3$,@#4         ; NO KT -- SET TRAP CATCHER.
011340  005002                  CLR     R2              ; FIRST ADDRESS (0).
011342  005003                  CLR     R3
011344  005712          2$:     TST     (R2)            ; SIZE FROM 0 UP...
011346  000240                  240
011350  062702  004000          ADD     #4000,R2        ;...IN 1K STEPS...
011354  005203                  INC     R3
011356  000772                  BR      2$              ;..UNTIL WE TRAP.
011360  162702  000002  3$:     SUB     #2,R2           ; R2 = LAST VIRTUAL ADDRESS.
011364  005001                  CLR     R1              ; R1 = 0 (PAF DOESN'T APPLY).
011366  000506                  BR      .SIZXIT         ; RETURN.
```

```
                                        .SIZKT: MOV     #KIPAR0,R1      ; 1ST "PAR" ADDRESS...
                                                MOV     #KIPDR0,R2      ; ...AND IT'S "PDR"...
                                                MOV     #8.,R3          ;...AND THERE ARE 8 OF EACH.
                                                CLR     R0              ; 1ST PAGE IS ZERO.
                                        1$:     MOV     R0,(R1)+        ; SET PAR'S = 0, 4K, 8K ... 28K.
                                                MOV     #77406,(R2)+    ; SET PDR'S = 4K, EX-UP, READ/WRITE.
                                                ADD     #200,R0         ; SET NEXT PAGE PAF (+4K)...
                                                SOB     R3,1$           ; ...AND LOOP UNTIL ALL LOADED.
                                                MOV     #177600,-(R1)   ; PAR7 IS THE I/O PAGE (PAF).
                                                CLR     -(R1)           ; PAR6 WILL DO THE SIZING.
                                                CLR     R3              ; R3 WILL COUNT THE K'S.

                                                MOV     #2$,@#4         ; TRAP TO 2$ IF NO 22 BIT SUPPORT.
                                                TST     MMR3            ; 22 BITS SUPPORTED ??
                                                MOV     #20,MMR3        ; MUST BE, SET 22 BIT MODE...
                                                MOV     #3$,@#4         ;...TRAP TO 3$ IF 22 BIT ADDRESS IS NXM.
                                                INC     MMR0            ;****** KT ON **************************
                                                MOV     #10000,(R1)     ; SET PAR6 AT START OF 22 BIT LAND...
                                                CMP     @#200,140200    ;...AND LOOK FOR WRAP-AROUND.
                                                BNE     3$              ; WE'RE HERE IF IT DIDN'T TRAP...
                                                                        ;...BRANCH IF IT DIDN'T WRAP-AROUND.
                                                CLR     MMR3            ; WRAP-AROUND -- MUST BE 18 BITS ONLY.
                                        2$:     MOV     #18.,..ABUSW    ; SET THE BUS WIDTH = 18.
                                                MOV     #7600,R4        ; SET SIZER LIMIT = 124K.
                                                BR      4$
                                        3$:     MOV     #22.,..ABUSW    ; WE HAVE A REAL 22 BIT ADDRESS SPACE.
                                                MOV     #170000,R4      ; SET SIZER LIMIT = 1920K.
                                        4$:
                                        .SIZMEM: MOV    #2$,@#4         ; TRAP TO 2$ WHEN DONE SIZING.
                                                CLR     (R1)            ; SET PAR6 AT 1ST PAGE (0).
                                        1$:     TST     140000          ; SIZE USING PAR6 (+0)...
                                                ADD     #40,(R1)        ;...IN 1K STEPS.
                                                INC     R3
                                                CMP     (R1),R4         ; REACHED LIMIT ??
                                                BLO     1$              ; LOOP IF NOT.
                                        2$:     MOV     (R1),R0         ; DONE, SAVE FINAL PAR6..;
                                                MOV     #1400,(R1)      ;...AND RESET IT TO BANK 6 (24K).
                                                CLR     MMR0            ;****** KT OFF **************************
                                                MOV     R0,R1           ; RECOVER SIZING RESULT.
                                                SUB     #40,R1          ; R1 = LAST 1K PAGE (PAF).
                                                MOV     #3776,R2        ; R2 = LAST ADDR IN THAT PAGE.

                                        .SIZXIT: MOV    R1,..LSTPG      ; RETURN LAST PAGE (PAF)...
                                                MOV     R2,..LSTAD      ;...LAST VIRTUAL ADDRESS...
                                                MOV     R3,..MSIZE      ;...AND TOTAL MEMORY SIZE (K).
                                                MOV     R5,SP           ; RECOVER OUR STACK POINTER...
                                                MOV     (SP)+,@#116     ;...AND THE ERROR VECTORS.
                                                MOV     (SP)+,@#114
                                                MOV     (SP)+,@#6
                                                MOV     (SP)+,@#4
                                                RTS     PC              ;...AND RETURN.
```

```
                                        .SBTTL  REPORT ENVIRONMENT
                                        ;
                                        ; SUBROUTINE TO DISPLAY THE CURRENT OPERATING ENVIRONMENT.
                                        ; CALLED ON EVERY START/RESTART, OR DIRECTLY VIA 'PRI'.
                                        ;
                                                BGNRPT
                                                                                        LSRPT::
                                        1$:     CLRB    MSGNOT
                                                TST     .LSTPG
                                                BNE     2$
                                                MOVB    #40,MSGNOT
                                        2$:     PRINTF  #MSG2,#MSGNOT   ; MMU
                                                                                        MOV     #MSGNOT,-(SP)
                                                                                        MOV     #MSG2,-(SP)
                                                                                        MOV     #2,-(SP)
                                                                                        MOV     SP,R0
                                                                                        TRAP    C$PNTF
                                                                                        ADD     #6,SP
                                                PRINTF  #MSG3,.MSIZE    ; MEM SIZE.
                                                                                        MOV     .MSIZE,-(SP)
                                                                                        MOV     #MSG3,-(SP)
                                                                                        MOV     #2,-(SP)
                                                                                        MOV     SP,R0
                                                                                        TRAP    C$PNTF
                                                                                        ADD     #6,SP
                                        3$:     PRINTF  #MSG4,.ABUSW    ; ADDRESS SPACE.
                                                                                        MOV     .ABUSW,-(SP)
                                                                                        MOV     #MSG4,-(SP)
                                                                                        MOV     #2,-(SP)
                                                                                        MOV     SP,R0
                                                                                        TRAP    C$PNTF
                                                                                        ADD     #6,SP
                                                EXIT    RPT
                                                                                        .WORD   J$JMP
                                                                                        .WORD   L10012-2-.
                                        MSG2:   .ASCIZ  '%N%A  MMU%T%A AVAILABLE'
                                        MSG3:   .ASCIZ  '%N%A  MEMORY SIZE %D4%A KW'
                                        MSG4:   .ASCIZ  '%N%A  %D2%A BIT ADDRESSING%N'
                                        MSGNOT: .ASCIZ  ' NOT'
                                                .EVEN
                                                ENDRPT
                                                                                L10012:
                                                                                        TRAP    C$RPT
```

```
                                        .SBTTL
                                        .SBTTL  RL DISKLESS CONTROLLER TESTS.
                                        .SBTTL
                                        .SBTTL    1 -- RLCS ADDRESSABILITY.

012114                                  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.
012114                                  STARS
                                        ;;*****************************************************************
012114                                          BEGIN.TEST
012114                                                                          T1::
012114  005037  010626              1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
012120  012777  177777  171176              MOV     #177777,@RLCS   ; WRITE RLCS.
012126  000240  000240                      240,240
012132  005777  171166                      TST     @RLCS           ; READ RLCS.
012136  000240  000240                      240,240
012142  005737  010626                      TST     TRPFLG          ; EITHER ONE TRAP ??
012146  001413                              BEQ     3$              ; NO, PROCEED.
012150  013737  003324  003432              MOV     RLCS,GDDAT      ;SET UP ERROR DATA
012156                                      SFERR   EM1,ERR1        ;BUS TIMEOUT IN ADDRESSING RLCS
012156  104454                                                              TRAP    C$ERSF
012160  000002                                                              .WORD   2
012162  030706                                                              .WORD   EM1
012164  026170                                                              .WORD   ERR1
012166                                  DODU    UNITST          ; DROP...
012166  013700  003322                                                      MOV     UNITST,R0
012172  104451                                                              TRAP    C$DODU
012174                                  DOCLN                   ;...AND ABORT.
012174  104444                                                              TRAP    C$DCLN
012176                          3$:     ENDTST
012176                                                                  L10013:
012176  104401                                                              TRAP    C$ETST
```

```
                                .SBTTL    2 -- RLBA ADDRESSABILITY.

                                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
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                        T2::
                                1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
                                        MOV     #177777,@RLBA   ; WRITE RLBA.
                                        240,240
                                        TST     @RLBA           ; READ RLBA.
                                        240,240
                                        TST     TRPFLG          ; ANY TRAPS ??
                                        BEQ     3$              ;NO, CONTINUE
                                        MOV     RLBA,GDDAT      ;SETUP ERROR DATA
                                        SFERR   EM2,ERR1        ;BUS TIMEOUT IN ADDRESSING RLBA
                                                                                TRAP    C$ERSF
                                                                                .WORD   3
                                                                                .WORD   EM2
                                                                                .WORD   ERR1
                                        DODU    UNITST          ; DROP...
                                                                                MOV     UNITST,R0
                                                                                TRAP    C$DODU
                                        DOCLN                   ;...AND ABORT.
                                                                                TRAP    C$DCLN
                                3$:     ENDTST
                                                                        L10014:
                                                                                TRAP    C$ETST
```

```
                                .SBTTL    3 -- RLDA ADDRESSABILITY.

                                STARS
                                ;;*****************************************************************
                                ;TEST TO SEE IF WE CAN ADDRESS THE DISK ADDRESS
                                ;REGISTER IF WE TRAP WE WILL REPORT THE ERROR
                                ;AND ABORT.  AFTER THIS TEST WE ONLY KNOW THAT
                                ;WE CAN ADDRESS THE REGISTER.
                                STARS
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                        T3::
                                1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
                                        MOV     #177777,@RLDA   ; WRITE RLDA
                                        240,240
                                        TST     @RLDA           ; READ RLDA
                                        240,240
                                        TST     TRPFLG          ; ANY TRAPS ??
                                        BEQ     3$              ;NO, CONTINUE
                                        MOV     RLDA,GDDAT      ;SETUP ERROR INFO
                                        SFERR   EM3,ERR1        ;BUS TIMEOUT IN ADDRESSING RLDA
                                                                                TRAP    C$ERSF
                                                                                .WORD   4
                                                                                .WORD   EM3
                                                                                .WORD   ERR1
                                        DODU    UNITST          ; DROP...
                                                                                MOV     UNITST,R0
                                                                                TRAP    C$DODU
                                        DOCLN                   ;...AND ABORT.
                                                                                TRAP    C$DCLN
                                3$:     ENDTST
                                                                        L10015:
                                                                                TRAP    C$ETST
```

```
1184
1185                                    .SBTTL    4 -- RLMP ADDRESSABILITY.
1186  012350                            STARS
  (2)                                   ;;*****************************************************************
1187                                    ;TEST TO SEE IF WE CAN ADDRESS THE MULTIPURPOSE
1188                                    ;REGISTER.  IF WE TRAP WE WILL REPORT THE ERROR AND
1189                                    ;ABORT.  AFTER THIS TEST WE ONLY KNOW THAT WE CAN
1190                                    ;ADDRESS THE REGISTER.
1191  012350                            STARS
  (2)                                   ;;*****************************************************************
1192  012350                                  BEGIN.TEST
  (4)  012350                                                                          T4::
1193  012350  005037  010626           1$:     CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
1194  012354  012777  177777  170750          MOV     #177777,@RLMP   ;WRITE RLMP
1195  012362  000240  000240                  240,240
1196  012366  005777  170740                  TST     @RLMP           ; READ RLMP
1197  012372  000240  000240                  240,240
1198  012376  005737  010626                  TST     TRPFLG          ; ANY TRAPS ??
1199  012402  001413                          BEQ     3$              ;NO, CONTINUE
1200  012404  013737  003332  003432          MOV     RLMP,GDDAT      ;SET UP ERROR INFO
1201  012412                                  SFERR   EM4,ERR1        ;BUS TIMEOUT IN ADDRESSING RLMP
  (5)  012412  104454                                                                  TRAP    C$ERSF
  (6)  012414  000005                                                                  .WORD   5
  (6)  012416  031005                                                                  .WORD   EM4
  (6)  012420  026170                                                                  .WORD   ERR1
1202  012422                                  DODU    UNITST          ; DROP...
  (3)  012422  013700  003322                                                          MOV     UNITST,R0
  (3)  012426  104451                                                                  TRAP    C$DODU
1203  012430                                  DOCLN                   ;...AND ABORT.
  (3)  012430  104444                                                                  TRAP    C$DCLN
1204  012432                          3$:     ENDTST
  (3)  012432                                                                  L10016:
  (3)  012432  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL    5 -- RLBAE ADDRESSABILITY (RLV12 ONLY).

                                STARS
                                ;;*****************************************************************
                                ; THE DEFAULT (SHIP) CONFIGURATION REQUIRES THAT THE
                                ; BAE REGISTER BE ENABLED.  CHECK IT AND REPORT ACCORDINGLY.
                                STARS
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                        T5::
012434  023727  003402  000002          CMP     RLTYP,#RLV12    ; RLV12 ??
012442  103444                        BLO     4$              ; NO, SKIP THIS TEST.
012444  005037  010626          1$:   CLR     TRPFLG          ;CLEAR TRAP OCCURANCE
012450  013737  003334  003432        MOV     RLBAE,GDDAT     ; SET ADDRESS IN CASE OF ERROR.
012456  012777  177777  170650        MOV     #-1,@RLBAE      ; WRITE BAE.
012464  000240  000240                240,240
012470  005777  170640                TST     @RLBAE          ; READ BAE.
012474  000240  000240                240,240
012500  023727  003402  000003        CMP     RLTYP,#RLV12X   ; DEFAULT CONFIG (W/BAE) ??
012506  001013                        BNE     3$              ; BR IF NOT.

012510  005737  010626          2$:   TST     TRPFLG          ; BAE SHOULD HAVE ANSWERED -- DID IT ??
012514  001417                        BEQ     4$              ; BR IF SO.
012516                                SFERR   EM4A,ERR1       ;BUS TIMEOUT IN ADDRESSING RLBAE
012516  104454                                                        TRAP    C$ERSF
012520  000006                                                        .WORD   6
012522  031032                                                        .WORD   EM4A
012524  026170                                                        .WORD   ERR1
012526                                  DODU    UNITST          ; DROP...
012526  013700  003322                                                MOV     UNITST,R0
012532  104451                                                        TRAP    C$DODU
012534                                  DOCLN                   ;...AND ABORT.
012534  104444                                                        TRAP    C$DCLN

012536  005737  010626          3$:   TST     TRPFLG          ; BAE SHOULD HAVE TRAPPED -- DID IT ??
012542  001004                        BNE     4$              ; BR IF SO.
012544                                DFERR   EM4B,ERR1       ; BAE IS NOT DISABLED.
012544  104455                                                        TRAP    C$ERDF
012546  000145                                                        .WORD   101
012550  031060                                                        .WORD   EM4B
012552  026170                                                        .WORD   ERR1
012554                          4$:   ENDTST
012554                                                          L10017:
012554  104401                                                        TRAP    C$ETST
```

```
                        .SBTTL    6 -- BUS RESET OF ALL REGISTERS.

012556                  STARS
  (2)                   ;;*****************************************************************
                        ;TEST THAT A BUS RESET WILL CLEAR THE PROPER BITS.
                        ;IN THE CONTROL AND STATUS REGISTER, THOSE BITS ARE
                        ;1-6,8,9,10,11,12,13,15.  BIT 15 WILL CLEAR ONLY
                        ;IF BIT 14 (DRIVE ERROR IS NOT SET).  BIT 0 (DRIVE READY)
                        ;IS A DON'T CARE.  IF AT THE START UP THIS TEST BIT
                        ;14 (DRIVE ERROR) IS SET WE WILL INSIST IF IS THERE AFTER
                        ;THE "RESET" ALONG WITH BIT 15 (COMPOSITE ERROR).  BITS
                        ;15-10 ARE NOT WRITEABLE.
                        ; THE OTHER REGISTERS SHOULD GO TO ZERO.
012556                  STARS
  (2)                   ;;*****************************************************************
012556                          BEGIN.TEST
  (4)  012556                                                                   T6::
012556                          SETPRI  #PRI07          ;PRIORITY TO SEVEN
  (3)  012556  012700  000340                                                   MOV     #PRI07,R0
  (3)  012562  104441                                                           TRAP    C$SPRI
012564  012777  000377  170532        MOV     #377,@RLCS      ;LOAD ALL RLCS LOADABLE BITS
012572  012737  000200  003432        MOV     #CRDY,GDDAT     ;SETUP EXPECTED CSR.
012600  032777  040000  170516        BIT     #DERR,@RLCS     ;DRIVE ERR SET?
012606  001403                        BEQ     1$              ;IF NOT DON'T EXPECT IT
012610  052737  140000  003432        BIS     #DERR!ERR,GDDAT  ;IT'S SET, INIT BETTER NOT CLR
012616  012777  177777  170502  1$:   MOV     #-1,@RLBA       ; LOAD ALL BITS IN THE OTHERS.
012624  012777  177777  170476        MOV     #-1,@RLDA
012632  023727  003402  000003        CMP     RLTYP,#RLV12X   ; RLV12X ??
012640  001003                        BNE     2$              ; NO, SKIP NEXT.
012642  012777  177777  170464        MOV     #-1,@RLBAE
012650  004537  010160          2$:   JSR     R5,WDELAY       ; DELAY BEFORE...
012654  000372                        250.
012656  000005                        RESET
012660  004537  010160          3$:   JSR     R5,WDELAY       ;...AND AFTER.
012664  000372                        250.
012666  017737  170432  003434        MOV     @RLCS,BDDAT     ;READ RLCS
012674  042737  000001  003434        BIC     #DRDY,BDDAT     ;CLEAR OUT DRDY - DON'T CARE
012702  023737  003434  003432        CMP     BDDAT,GDDAT     ;DID INIT WORK ON CSR.
012710  001404                        BEQ     4$              ;YES, BRANCH
012712                                DFERR   EM69,ERR2       ;BUS-INIT FAILED ON CSR.
  (5)  012712  104455                                                           TRAP    C$ERDF
  (6)  012714  000146                                                           .WORD   102
  (6)  012716  033311                                                           .WORD   EM69
  (6)  012720  026220                                                           .WORD   ERR2
012722  005037  003432          4$:   CLR     GDDAT           ; EXPECT 0 IN THE OTHERS.
012726  017737  170374  003434        MOV     @RLBA,BDDAT
012734  001404                        BEQ     5$
012736                                DFERR   EM70,ERR2       ; BUS-INIT FAILED ON RLBA.
  (5)  012736  104455                                                           TRAP    C$ERDF
  (6)  012740  000147                                                           .WORD   103
  (6)  012742  033345                                                           .WORD   EM70
  (6)  012744  026220                                                           .WORD   ERR2
012746  017737  170356  003434  5$:   MOV     @RLDA,BDDAT
012754  001404                        BEQ     6$
012756                                DFERR   EM71,ERR2       ; BUS-INIT FAILED ON RLDA.
  (5)  012756  104455                                                           TRAP    C$ERDF
```

```
                                                                                        .WORD   104
                                                                                        .WORD   EM71
                                                                                        .WORD   ERR2
1277  012766  023727  003402  000003  6$:     CMP     RLTYP,#RLV12X   ; RLV12X ??
1278  012774  001010                          BNE     7$              ; WE'RE DONE IF NOT.
1279  012776  017737  170332  003434          MOV     @RLBAE,BDDAT
1280  013004  001404                          BEQ     7$
1281  013006                                  DFERR   EM71A,ERR2      ; BUS-INIT FAILED ON RLBAE.
(5)   013006  104455                                                                  TRAP    C$ERDF
(6)   013010  000151                                                                  .WORD   105
(6)   013012  033435                                                                  .WORD   EM71A
(6)   013014  026220                                                                  .WORD   ERR2
1282  013016                          7$:     ENDTST
(3)   013016                                                                L10020:
(3)   013016  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL    7 -- READ WRITE OF RLCS.

013020                          STARS
                                ;;*************************************************************************
                                ;TEST THAT WE CAN WRITE/READ BITS 8,9 AND BITS 6-1
                                ;OF THE CONTROL AND STATUS REGISTER.  BITS 15-10 AND 0
                                ;ARE DON'T CARE BITS AT THIS TIME AND BIT 7
                                ;(CONTROLLER READY) IS ALWAYS WRITTEN TO A ONE.
013020                          STARS
                                ;;*************************************************************************
013020                                  BEGIN.TEST
013020                                                                          T7::
013020  012703  004100                  MOV     #CSPAT,R3       ;SET UP TABLE POINTER OF PATTERNS
013024                                  BGNSEG                  ;****START OF SEGMENT****
013024  104404                                                                  TRAP    C$BSEG
013026  011337  003432          1$:     MOV     (R3),GDDAT      ;GET PATTERN INTO GDDAT
013032  052737  000200  003432          BIS     #200,GDDAT      ;INSURE GO IS SET
013040  013777  003432  170256          MOV     GDDAT,@RLCS     ;LOAD RLCS (CONTROL AND STATUS)
013046  032777  040000  170250          BIT     #DERR,@RLCS     ;IF DRIVE ERROR PRESENT
013054  001403                          BEQ     2$              ;THEN EXPECT DRIVE AND
013056  052737  140000  003432          BIS     #ERR!DERR,GDDAT ;COMPOSITE ERROR
013064  017737  170234  003434  2$:     MOV     @RLCS,BDDAT     ;READ RLCS BACK
013072  042737  000001  003434          BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
013100  023737  003432  003434          CMP     GDDAT,BDDAT     ;DID WE READ WHAT WE LOADED
013106  001404                          BEQ     3$              ;YES, THEN BRANCH
013110                                  DFERR   EM5,ERR2        ;WRONG DATA IN RLCS
013110  104455                                                                  TRAP    C$ERDF
013112  000152                                                                  .WORD   106
013114  031127                                                                  .WORD   EM5
013116  026220                                                                  .WORD   ERR2
013120                          3$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA:LOE).
013120  104410                                                                  TRAP    C$ESCAPE
013122  000012                                                                  .WORD   10000$-.
013124  005723                          TST     (R3)+           ;BUMP FOR NEXT PATTERN
013126  020327  004176                  CMP     R3,#CSEND       ;CHECK FOR END
013132  001335                          BNE     1$              ;NOT END, LOAD NEXT PATTERN
013134                                  ENDSEG                  ;****END OF SEGMENT****
013134                                                                  10000$:
013134  104405                                                          TRAP    C$ESEG
013136                                  ENDTST
013136                                                                  L10021:
013136  104401                                                          TRAP    C$ETST
```

```
                                                                    D 5
                        .SBTTL    8 -- READ WRITE OF RLBA.

013140                  STARS
                        ;;*****************************************************************
                        ;TEST THAT WE CAN WRITE/READ BITS 15 THRU 1 OF THE
                        ;BUS ADDRESS REGISTER.  FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
                        ;GROWING 0 AND SHIFTING 0.  BIT 0 IS ALSO LOADED BUT
                        ;SHOULD ALWAYS COME BACK AS 0
013140                  STARS
                        ;;*****************************************************************
013140                          BEGIN.TEST
013140                                                                  T8::
013140  012703  003670          MOV     #BEGPAT,R3      ;GET START OF PATTERN LIST
013144                          BGNSEG                  ;****START OF SEGMENT****
013144  104404                                                  TRAP    C$BSEG
013146  011337  003432  1$:     MOV     (R3),GDDAT      ;GET PATTERN TO SEND
013152  023727  003402  000000  CMP     RLTYP,#RL11     ; RL11 ??
013160  001003                  BNE     2$              ;NO.
013162  042737  000001  003432  BIC     #BIT0,GDDAT     ;YES, KEEP RLBA EVEN (UNIBUS)
013170  013777  003432  170130  2$:     MOV     GDDAT,@RLBA     ;LOAD PATTERN TO BUS ADDRESS
013176  017737  170124  003434  MOV     @RLBA,BDDAT     ;READ IT BACK
013204  023737  003432  003434  CMP     GDDAT,BDDAT     ;IS IT CORRECT?
013212  001404                  BEQ     3$              ;IF SO, BRANCH
013214                          DFERR   EM6,ERR2        ;DATA WRONG IN RLBA
013214  104455                                                  TRAP    C$ERDF
013216  000153                                                  .WORD   107
013220  031200                                                  .WORD   EM6
013222  026220                                                  .WORD   ERR2
013224  3$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA;LOE).
013224  104410                                                  TRAP    C$ESCAPE
013226  000012                                                  .WORD   10000$-.
013230  005723          TST     (R3)+           ;BUMP FOR NEXT PATTERN
013232  020327  004076  CMP     R3,#ENDPAT      ;CHECK FOR END
013236  001343          BNE     1$              ; LOOP TIL DONE.
013240                  ENDSEG                  ;****END OF SEGMENT****
013240                                                  10000$:
013240  104405                                          TRAP    C$ESEG
013242                  ENDTST
013242                                                  L10022:
013242  104401                                          TRAP    C$ETST
```

```
                                                                E 5
1340                                    .SBTTL    9 -- READ WRITE OF RLDA.
1341
1342  013244                            STARS
  (2)                                   ;;*****************************************************************
1343                                    ;TEST THAT WE CAN WRITE/READ THE DISK ADDRESS REGISTER
1344                                    ;ALL BIT POSITIONS ARE WRITTEN USING FOUR PATTERNS:
1345                                    ;GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
1346  013244                            STARS
  (2)                                   ;;*****************************************************************
1347  013244                                    BEGIN.TEST
  (4)   013244                                                                      T9::
1348  013244  012703  003670                   MOV     #BEGPAT,R3      ;SET UP POINTER TO PATTERN LIST
1349  013250                                    BGNSEG                  ;****START OF SEGMENT****
  (3)   013250  104404                                                              TRAP    C$BSEG
1350  013252  011337  003432           1$:     MOV     (R3),GDDAT      ;GET PATTERN
1351  013256  013777  003432  170044           MOV     GDDAT,@RLDA     ;LOAD PATTERN IN DA
1352  013264  017737  170040  003434           MOV     @RLDA,BDDAT     ;READ PATTERN BACK
1353  013272  023737  003432  003434           CMP     GDDAT,BDDAT     ;IS IT CORRECT?
1354  013300  001404                           BEQ     2$              ;BRANCH IF CORRECT
1355  013302                                    DFERR   EM7,ERR2        ;WRONG DATA IN RLDA
  (5)   013302  104455                                                              TRAP    C$ERDF
  (6)   013304  000154                                                              .WORD   108
  (6)   013306  031226                                                              .WORD   EM7
  (6)   013310  026220                                                              .WORD   ERR2
1356  013312                            2$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA:LOE).
  (3)   013312  104410                                                              TRAP    C$ESCAPE
  (3)   013314  000012                                                              .WORD   10000$-.
1357  013316  005723                           TST     (R3)+           ;BUMP POINTER
1358  013320  020327  004076                   CMP     R3,#ENDPAT      ;AT END OF PATTERNS?
1359  013324  001352                           BNE     1$              ;NO, BRANCH BACK
1360  013326                                    ENDSEG                  ;****END OF SEGMENT****
  (3)   013326                                                                      10000$:
  (3)   013326  104405                                                              TRAP    C$ESEG
1361  013330                                    ENDTST
  (3)   013330                                                          L10023:
  (3)   013330  104401                                                              TRAP    C$ETST
```

```
                                .SBTTL   10 -- READ WRITE OF RLBAE (RLV12 ONLY).

013332                          STARS
                                ;;*************************************************************************
                                ;TEST THAT WE CAN WRITE/READ THE BUS ADDRESS EXTENSION REGISTER.
                                ;ALL BIT POSITIONS ARE WRITTEN USING FOUR PATTERNS:
                                ;GROWING 1, SHIFTING 1, GROWING 0 AND SHIFTING 0
                                ; ONLY THE LOW 6 BITS ARE TESTED RLBAE<5:0>
013332                          STARS
                                ;;*************************************************************************
013332                                  BEGIN.TEST
013332                                                                          T10::
013332  023727  003402  000003         CMP     RLTYP,#RLV12X   ; BAE ENABLED ??
013340  001035                         BNE     3$              ; NO, DON'T EVEN TRY.
013342  012703  003670                 MOV     #BEGPAT,R3      ;SET UP POINTER TO PATTERN LIST
013346                                  BGNSEG                  ;****START OF SEGMENT****
013346  104404                                                                  TRAP    C$BSEG
013350  011337  003432         1$:     MOV     (R3),GDDAT      ;GET PATTERN
013354  013777  003432  167752         MOV     GDDAT,@RLBAE    ;LOAD PATTERN IN BAE
013362  017737  167746  003434         MOV     @RLBAE,BDDAT    ;READ PATTERN BACK
013370  042737  177700  003432         BIC     #^C77,GDDAT     ; ** ONLY 5:0 ARE VALID BITS.
013376  023737  003432  003434         CMP     GDDAT,BDDAT     ;IS IT CORRECT?
013404  001404                         BEQ     2$              ;BRANCH IF CORRECT
013406                                  DFERR   EM8,ERR2        ;WRONG DATA IN RLBAE
013406  104455                                                                  TRAP    C$ERDF
013410  000155                                                                  .WORD   109
013412  031254                                                                  .WORD   EM8
013414  026220                                                                  .WORD   ERR2
013416                          2$:     ESCAPE SEG              ; ESCAPE AND LOOP (IF FLA;LOE).
013416  104410                                                                  TRAP    C$ESCAPE
013420  000012                                                                  .WORD   10000$-.
013422  005723                  1$:     TST     (R3)+           ;BUMP POINTER
013424  020327  004076                 CMP     R3,#ENDPAT      ;AT END OF PATTERNS?
013430  001347                         BNE     1$              ;NO, BRANCH BACK
013432                                  ENDSEG                  ;****END OF SEGMENT****
013432                                                                  10000$:
013432  104405                                                          TRAP    C$ESEG
013434                          3$:     ENDTST
013434                                                                  L10024:
013434  104401                                                          TRAP    C$ETST
```

```
                                                    G 5
1390                                .SBTTL   11 -- BIS AND BIC OF RLCS.
1391
1392  013436                        STARS
 (2)                                ;;*****************************************************************
1393                                ;TEST THAT WE CAN USE THE "BIS" AND "BIC" INSTRUCTIONS ON THE CONTROL
1394                                ;AND STATUS REGISTER.  BITS 8,9 AND 6-1 ARE TESTED TO
1395                                ;SET INDIVIDUALLY AS WELL AS COLLECTIVELY WITHOUT DESTROYING
1396                                ;ANY PREVIOUS DATA PATTERN
1397  013436                        STARS
 (2)                                ;;*****************************************************************
1398  013436                            BEGIN.TEST
 (4)  013436                                                                  T11::
1399  013436  012703  004100                MOV     #CSPAT,R3       ;GET BEGINNING OF LIST
1400  013442                                BGNSEG                  ;***** START SEGMENT *****
 (3)  013442  104404                                                          TRAP    C$BSEG
1401  013444  012777  000200  167652  1$:   MOV     #CRDY,@RLCS     ;INSURE GO IS THERE
1402  013452  011337  003432                MOV     (R3),GDDAT      ;SET UP EXPECTED RLCS...
1403  013456  052737  000200  003432        BIS     #CRDY,GDDAT     ;...IN GDDAT.
1404  013464  051377  167634                BIS     (R3),@RLCS      ; *** BIT SET PATTERN IN RLCS
1405  013470  032777  040000  167626        BIT     #DERR,@RLCS     ;IF ERROR BIT SET THEN...
1406  013476  001403                        BEQ     2$              ;...EXPECT IT ON THE READ BACK.
1407  013500  052737  140000  003432        BIS     #ERR!DERR,GDDAT
1408  013506  017737  167612  003434  2$:   MOV     @RLCS,BDDAT     ;READ RLCS TO CHECK "BIS"
1409  013514  042737  000001  003434        BIC     #DRDY,BDDAT     ;CLEAR OUT DRIVE READY
1410  013522  023737  003434  003432        CMP     BDDAT,GDDAT     ;DID BIS WORK?
1411  013530  001404                        BEQ     3$              ;BRANCH IF OKAY
1412  013532                                DFERR   EM61,ERR2       ;WRONG DATA IN RLCS ON BIS.
 (5)  013532  104455                                                          TRAP    C$ERDF
 (6)  013534  000156                                                          .WORD   110
 (6)  013536  032617                                                          .WORD   EM61
 (6)  013540  026220                                                          .WORD   ERR2
1413  013542                        3$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA;LOE).
 (3)  013542  104410                                                          TRAP    C$ESCAPE
 (3)  013544  000114                                                          .WORD   10000$-.
1414  013546  012777  001776  167550  4$:   MOV     #1776,@RLCS     ;SET ALL SETTABLE BITS
1415  013554  012737  001776  003432        MOV     #1776,GDDAT     ;SET UP EXPECT DATA IN...
1416  013562  041337  003432                BIC     (R3),GDDAT      ;...GDDAT
1417  013566  041377  167532                BIC     (R3),@RLCS      ; *** XCT BIC ON RLCS.
1418  013572  032777  040000  167524        BIT     #DERR,@RLCS     ;IF DRIVE ERROR BIT SET...
1419  013600  001403                        BEQ     5$              ;...EXPECT IT SET WHEN WE READ.
1420  013602  052737  140000  003432        BIS     #ERR!DERR,GDDAT
1421  013610  017737  167510  003434  5$:   MOV     @RLCS,BDDAT     ;MOVE RLCS TO BDDAT FOR COMPARE
1422  013616  042737  000001  003434        BIC     #DRDY,BDDAT     ;CLEAR DRIVE READY
1423  013624  023737  003434  003432        CMP     BDDAT,GDDAT     ;DID "BIC" WORK PROPERLY
1424  013632  001404                        BEQ     6$              ;BRANCH IF OKAY
1425  013634                                DFERR   EM62,ERR2       ;WRONG DATA IN RLCS ON BIC.
 (5)  013634  104455                                                          TRAP    C$ERDF
 (6)  013636  000157                                                          .WORD   111
 (6)  013640  032665                                                          .WORD   EM62
 (6)  013642  026220                                                          .WORD   ERR2
1426  013644                        6$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA;LOE).
 (3)  013644  104410                                                          TRAP    C$ESCAPE
 (3)  013646  000012                                                          .WORD   10000$-.
1427  013650  005723                        TST     (R3)+           ;GET NEXT PATTERN
1428  013652  020327  004176                CMP     R3,#CSEND       ;AT END OF LIST
1429  013656  001272                        BNE     1$              ;NO, GO BACK WITH NEXT PATTERN
```

```
                                                                    H 5
 1430  013660                        ENDSEG
   (3) 013660
   (3) 013660  104405                                   10000$:
                                                                TRAP    C$ESEG
 1431  013662                        ENDTST
   (3) 013662                                           L10025:
   (3) 013662  104401                                           TRAP    C$ETST
```

```
                                                                                                                    I 5
                                        .SBTTL   12 -- BIS AND BIC OF RLBA.

                                        STARS
                                        ;;*****************************************************************
                                        ;TEST THAT THE "BIS" AND "BIC" INSTRUCTIONS WILL WORK ON THE
                                        ;BUS ADDRESS REGISTER.  BITS 15-0 ARE LOADED, ONLY BITS 15-1
                                        ;ARE EXPECTED BACK.  FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
                                        ;GROWING 0, AND SHIFTING 0.
                                        STARS
                                        ;;*****************************************************************
                                                BEGIN.TEST
                                                                                                        T12::
013664  012703  003670                          MOV     #BEGPAT,R3      ;GET START OF LIST
013670                                          BGNSEG                  ;**** START SEGMENT ****
(3) 013670  104404                                                                      TRAP    C$BSEG
013672  005077  167430          1$:             CLR     @RLBA           ;CLEAR "BA"
013676  011337  003432                          MOV     (R3),GDDAT      ;SET EXPECTED
013702  023727  003402  000000                  CMP     RLTYP,#RL11     ; RL11 ??
013710  001003                                  BNE     2$              ; NO.
013712  042737  000001  003432                  BIC     #1,GDDAT        ;BIT 0 CAN'T SET IN RLBA (UNIBUS)
013720  051377  167402          2$:             BIS     (R3),@RLBA      ; XCT BIS RLBA WITH PATTERN
013724  017737  167376  003434                  MOV     @RLBA,BDDAT     ;READ "BA"
013732  023737  003434  003432                  CMP     BDDAT,GDDAT     ;DID RLBA LOAD PROPERLY?
013740  001404                                  BEQ     3$              ;BRANCH IF YES
013742                                          DFERR   EM63,ERR2       ;WRONG DATA IN RLBA ON BIS.
(5) 013742  104455                                                                      TRAP    C$ERDF
(6) 013744  000160                                                                      .WORD   112
(6) 013746  032735                                                                      .WORD   EM63
(6) 013750  026220                                                                      .WORD   ERR2
013752                                  3$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA;LOE).
(3) 013752  104410                                                                      TRAP    C$ESCAPE
(3) 013754  000070                                                                      .WORD   10000$-.
013756  012777  177776  167342      4$:     MOV     #-2,@RLBA       ;SET RLBA TO ALL 1'S (BIT 0=0)
013764  012737  177776  003432              MOV     #-2,GDDAT       ;SET UP EXPECTED RESULTS
013772  041337  003432                      BIC     (R3),GDDAT      ;IN GDDAT
013776  041377  167324                      BIC     (R3),@RLBA      ; XCT BIC RLBA
014002  017737  167320  003434              MOV     @RLBA,BDDAT     ;READ RLBA
014010  023737  003434  003432              CMP     BDDAT,GDDAT     ;BIC WORK OKAY?
014016  001404                              BEQ     5$              ;IF YES BRANCH
014020                                      DFERR   EM64,ERR2       ;WRONG DATA IN RLBA ON BIC.
(5) 014020  104455                                                                      TRAP    C$ERDF
(6) 014022  000161                                                                      .WORD   113
(6) 014024  033003                                                                      .WORD   EM64
(6) 014026  026220                                                                      .WORD   ERR2
014030                                  5$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA:LOE).
(3) 014030  104410                                                                      TRAP    C$ESCAPE
(3) 014032  000012                                                                      .WORD   10000$-.
014034  005723                          TST     (R3)+           ;GET NEXT PATTERN
014036  020327  004076                  CMP     R3,#ENDPAT      ;HAVE WE COMPLETED LIST
014042  001313                          BNE     1$              ;NO, GO BACK FOR NEXT
014044                                  ENDSEG
(3) 014044                                                                      10000$:
(3) 014044  104405                                                                      TRAP    C$ESEG
014046                                  ENDTST
(3) 014046                                                                      L10026:
(3) 014046  104401                                                                      TRAP    C$ETST
```

```
                                                                                                                    J 5
                                .SBTTL   13 -- BIS AND BIC OF RLDA.

                                STARS
                                ;;*****************************************************************
                                ;TEST THAT THE "BIS" AND "BIC" INSTRUCTIONS WILL WORK ON THE DISK
                                ;ADDRESS REGISTER.  BITS 15-0 ARE TESTED WITH 4 PATTERNS, GROWING 1,
                                ;SHIFTING 1, GROWING 0, AND SHIFTING 0.
                                STARS
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                                T13::
                                        MOV     #BEGPAT,R3      ;GET START OF LIST
                                        BGNSEG                  ; **** START SEGMENT ****
                                                                                        TRAP    C$BSEG
                                1$:     CLR     @RLDA           ;CLEAR "DA"
                                        MOV     (R3),GDDAT      ;SET EXPECTED
                                        BIS     (R3),@RLDA      ; XCT BIS RLDA
                                        MOV     @RLDA,BDDAT     ;READ RLDA
                                        CMP     BDDAT,GDDAT     ;IS RLDA CORRECT
                                        BEQ     2$              ;IF OKAY BRANCH
                                        DFERR   EM65,ERR2       ;WRONG DATA IN RLDA ON BIS.
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   114
                                                                                        .WORD   EM65
                                                                                        .WORD   ERR2
                                2$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA;LOE).
                                                                                        TRAP    C$ESCAPE
                                                                                        .WORD   10000$-.
                                3$:     MOV     #-1,@RLDA       ;SET RLDA TO ALL 1'S
                                        MOV     #-1,GDDAT       ;SET EXPECTED DATA
                                        BIC     (R3),GDDAT      ;SET EXPECTED DATA
                                        BIC     (R3),@RLDA      ; XCT "BIC" RLDA
                                        MOV     @RLDA,BDDAT     ;READ RLDA
                                        CMP     GDDAT,BDDAT     ;DID "BIC" WORK?
                                        BEQ     4$              ;IF IT DID BRANCH
                                        DFERR   EM66,ERR2       ;WRONG DATA IN RLDA ON BIC.
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   115
                                                                                        .WORD   EM66
                                                                                        .WORD   ERR2
                                4$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA;LOE).
                                                                                        TRAP    C$ESCAPE
                                                                                        .WORD   10000$-.
                                        TST     (R3)+           ;GET NEXT PATTERN
                                        CMP     R3,#ENDPAT      ;DONE?
                                        BNE     1$              ;NO GO BACK
                                        ENDSEG
                                                                                10000$:
                                                                                        TRAP    C$ESEG
                                        ENDTST
                                                                                L10027:
                                                                                        TRAP    C$ETST
```

```
                                .SBTTL   14 -- BIS AND BIC OF RLBAE (RLV12 ONLY).

014216                          STARS
                                ;;*****************************************************************
                                ;TEST THAT THE "BIS" AND "BIC" INSTRUCTIONS WILL WORK ON THE
                                ;BUS ADDRESS EXTENSION REGISTER.  ALL BITS ARE LOADED BUT ONLY <5:0>
                                ;ARE EXPECTED BACK.  FOUR PATTERNS ARE USED: GROWING 1, SHIFTING 1,
                                ;GROWING 0, AND SHIFTING 0.
014216                          STARS
                                ;;*****************************************************************
014216                                  BEGIN.TEST
014216                                                                          T14::
014216  023727  003402  000003         CMP     RLTYP,#RLV12X   ; BAE ENABLED ??
014224  001070                          BNE     5$              ; SKIP IT IF NOT.
014226  012703  003670                  MOV     #BEGPAT,R3      ;GET START OF LIST
014232                                  BGNSEG                  ; **** START SEGMENT ****
014232  104404                                                                  TRAP    C$BSEG
014234  005077  167074          1$:     CLR     @RLBAE          ;CLEAR "BAE"
014240  011337  003432                  MOV     (R3),GDDAT      ;SET EXPECTED...
014244  042737  177700  003432         BIC     #^C77,GDDAT     ;...ONLY <5:0> ARE VALID.
014252  051377  167056                  BIS     (R3),@RLBAE     ; XCT BIS RLBAE WITH PATTERN
014256  017737  167052  003434         MOV     @RLBAE,BDDAT    ;READ "BAE"
014264  023737  003434  003432         CMP     BDDAT,GDDAT     ;DID IT LOAD PROPERLY?
014272  001404                          BEQ     2$              ;BRANCH IF YES
014274                                  DFERR   EM67,ERR2       ;WRONG DATA IN RLBAE ON BIS.
014274  104455                                                                  TRAP    C$ERDF
014276  000164                                                                  .WORD   116
014300  033171                                                                  .WORD   EM67
014302  026220                                                                  .WORD   ERR2
014304                          2$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA;LOE).
014304  104410                                                                  TRAP    C$ESCAPE
014306  000076                                                                  .WORD   10000$-.
014310  012777  177777  167016 3$:     MOV     #-1,@RLBAE      ;SET RLBAE TO ALL 1'S
014316  012737  177777  003432         MOV     #-1,GDDAT       ;SET UP EXPECTED RESULTS...
014324  041337  003432                  BIC     (R3),GDDAT      ;...IN GDDAT...
014330  042737  177700  003432         BIC     #^C77,GDDAT     ;...<5:0> ONLY.
014336  041377  166772                  BIC     (R3),@RLBAE     ; XCT BIC RLBAE
014342  017737  166766  003434         MOV     @RLBAE,BDDAT    ;READ RLBAE
014350  023737  003434  003432         CMP     BDDAT,GDDAT     ;BIC WORK OKAY?
014356  001404                          BEQ     4$              ;IF YES BRANCH
014360                                  DFERR   EM68,ERR2       ;WRONG DATA IN RLBAE ON BIC.
014360  104455                                                                  TRAP    C$ERDF
014362  000165                                                                  .WORD   117
014364  033240                                                                  .WORD   EM68
014366  026220                                                                  .WORD   ERR2
014370                          4$:     ESCAPE  SEG             ; ESCAPE AND LOOP (IF FLA;LOE).
014370  104410                                                                  TRAP    C$ESCAPE
014372  000012                                                                  .WORD   10000$-.
014374  005723                          TST     (R3)+           ;GET NEXT PATTERN
014376  020327  004076                  CMP     R3,#ENDPAT      ;HAVE WE COMPLETED LIST
014402  001314                          BNE     1$              ;NO, GO BACK FOR NEXT
014404                                  ENDSEG
014404                                                                  10000$:
014404  104405                                                                  TRAP    C$ESEG
014406                          5$:     ENDTST
014406                                                                  L10030:
```

```
                                                                    L 5
   (3)  014406  104401                                                  TRAP    CSETST
```

```
                                .SBTTL   15 -- UNIQUENESS OF RLCS.

014410                          STARS
                                ;;*****************************************************************
                                ;TEST THE UNIQUENESS OF THE CONTROL AND STATUS
                                ;REGISTER.  THE RLBA AND RLDA ARE PRELOADED WITH
                                ;177776 AND 177777 RESPECTIVELY.  THE RLCS IS THEN
                                ;LOADED TO INSURE THAT NEITHER THE RLBA OR RLDA
                                ;ARE MODIFIED BY THE WRITING OF THE RLCS.
                                ; IF RLV12, CHECK THE BAE ALSO, NOTING THAT CSR<5:4> SHOULD
                                ; BE MIRRORED IN BAE<1:0>.
014410                          STARS
                                ;;*****************************************************************
014410                                  BEGIN.TEST
014410                                                                          T15::
014410  012777  177776  166710        MOV     #-2,@RLBA               ;SET RLBA TO ALL 1'S
014416  012777  177777  166704        MOV     #-1,@RLDA               ;SET RLDA TO ALL 1'S
014424  023727  003402  000003        CMP     RLTYP,#RLV12X
014432  001003                        BNE     1$
014434  012777  000077  166672        MOV     #77,@RLBAE              ; SET BAE TO ALL 1'S.
014442  012777  000200  166654  1$:   MOV     #CRDY,@RLCS             ;WRITE RLCS
014450  012737  177776  003432        MOV     #-2,GDDAT       ;SET UP EXPECTED BA.
014456  017737  166644  003434        MOV     @RLBA,BDDAT     ;READ RLBA
014464  023737  003432  003434        CMP     GDDAT,BDDAT
014472  001405                        BEQ     2$              ; BR IF BA UNCHANGED.
014474                                DFERR   EM72,ERR2       ;CS MODIFIED BA
014474  104455                                                                    TRAP    C$ERDF
014476  000166                                                                    .WORD   118
014500  033472                                                                    .WORD   EM72
014502  026220                                                                    .WORD   ERR2
014504                                CKLOOP
014504  104406                                                                    TRAP    C$CLP1
014506  012737  177777  003432  2$:   MOV     #-1,GDDAT       ;SET UP EXPECTED DA.
014514  017737  166610  003434        MOV     @RLDA,BDDAT     ;READ DA
014522  023737  003432  003434        CMP     GDDAT,BDDAT
014530  001405                        BEQ     3$              ; BR IF DA UNCHANGED.
014532                                DFERR   EM73,ERR2       ;CS MODIFIED DA
014532  104455                                                                    TRAP    C$ERDF
014534  000167                                                                    .WORD   119
014536  033525                                                                    .WORD   EM73
014540  026220                                                                    .WORD   ERR2
014542                                CKLOOP
014542  104406                                                                    TRAP    C$CLP1
014544  023727  003402  000003  3$:   CMP     RLTYP,#RLV12X
014552  001016                        BNE     4$
014554  012737  000074  003432        MOV     #74,GDDAT       ; SET EXPECTED BAE.
014562  017737  166546  003434        MOV     @RLBAE,BDDAT    ; READ IT.
014570  023737  003432  003434        CMP     GDDAT,BDDAT
014576  001404                        BEQ     4$              ; BR IF BAE IS RIGHT.
014600                                DFERR   EM73A,ERR2      ; BAE WRONG AFTER WRITING CS.
014600  104455                                                                    TRAP    C$ERDF
014602  000170                                                                    .WORD   120
014604  033560                                                                    .WORD   EM73A
014606  026220                                                                    .WORD   ERR2
014610                          4$:   ENDTST
014610                                                                  L10031:
```

SEQ 0065

(3) 014610 104401 TRAP CSETST

```
                                        .SBTTL   16 -- UNIQUENESS OF RLBA.

014612                                  STARS
  (2)                                   ;;*****************************************************************
                                        ;
                                        ;TEST THE UNIQUENESS OF THE BUS ADDRESS REGISTER.  THE
                                        ;RLCS AND RLDA ARE LOADED WITH XXX20X AND 177777
                                        ;RESPECTIVELY.  THE RLBA IS THEN WRITTEN TO INSURE
                                        ;THAT NEITHER THE RLCS OR RLDA ARE MODIFIED.
                                        ; IF RLV12, CHECK THAT BAE IS UNAFFECTED ALSO.
014612                                  STARS
                                        ;;*****************************************************************
014612                                          BEGIN.TEST
014612                                                                          T16::
014612  012737  000200  003432                 MOV     #CRDY,GDDAT     ;CONTROLLER READY
014620  032777  040000  166476                 BIT     #DERR,@RLCS     ;IF DRIVE ERROR IS SET...
014626  001403                                 BEQ     1$
014630  052737  140000  003432                 BIS     #ERR!DERR,GDDAT ;...EXPECT IT BACK.
014636  013777  003432  166460         1$:     MOV     GDDAT,@RLCS     ;LOAD RLCS
014644  012777  177777  166456                 MOV     #-1,@RLDA       ;LOAD RLDA
014652  023727  003402  000003                 CMP     RLTYP,#RLV12X
014660  001003                                 BNE     2$
014662  012777  000074  166444                 MOV     #74,@RLBAE      ;LOAD RLBAE.
014670  005077  166432                 2$:     CLR     @RLBA           ; WRITE TO RLBA
014674  017737  166424  003434                 MOV     @RLCS,BDDAT     ;READ RLCS
014702  042737  000001  003434                 BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
014710  023737  003432  003434                 CMP     GDDAT,BDDAT
014716  001405                                 BEQ     3$              ; BR IF CS UNCHANGED.
014720                                         DFERR   EM74,ERR2       ;BA MODIFIED CS
  (5)   014720  104455                                                          TRAP    C$ERDF
  (6)   014722  000171                                                          .WORD   121
  (6)   014724  033617                                                          .WORD   EM74
  (6)   014726  026220                                                          .WORD   ERR2
1605    014730                                  CKLOOP
  (3)   014730  104406                                                          TRAP    C$CLP1
1606    014732  012737  177777  003432 3$:     MOV     #-1,GDDAT       ; SET EXPECTED DA.
1607    014740  017737  166364  003434         MOV     @RLDA,BDDAT     ; READ IT.
1608    014746  023737  003432  003434         CMP     GDDAT,BDDAT
1609    014754  001405                         BEQ     4$              ; BR IF DA UNCHANGED.
1610    014756                                 DFERR   EM75,ERR2       ;BA MODIFIED DA
  (5)   014756  104455                                                          TRAP    C$ERDF
  (6)   014760  000172                                                          .WORD   122
  (6)   014762  033651                                                          .WORD   EM75
  (6)   014764  026220                                                          .WORD   ERR2
1611    014766                                  CKLOOP
  (3)   014766  104406                                                          TRAP    C$CLP1
1612    014770  023727  003402  000003 4$:     CMP     RLTYP,#RLV12X
1613    014776  001016                         BNE     5$
1614    015000  012737  000074  003432         MOV     #74,GDDAT       ; SET EXPECTED BAE.
1615    015006  017737  166322  003434         MOV     @RLBAE,BDDAT    ; READ IT.
1616    015014  023737  003432  003434         CMP     GDDAT,BDDAT
1617    015022  001404                         BEQ     5$              ; BR IF BAE UNCHANGED.
1618    015024                                 DFERR   EM75A,ERR2      ; BA MODIFIED BAE
  (5)   015024  104455                                                          TRAP    C$ERDF
  (6)   015026  000173                                                          .WORD   123
  (6)   015030  033703                                                          .WORD   EM75A
  (6)   015032  026220                                                          .WORD   ERR2
```

```
                                                                C 6
 1619  015034                        5$:     ENDTST
   (3)  015034
   (3)  015034  104401                                  L10032:
                                                                TRAP    CSETST
```

```
                                .SBTTL   17 -- UNIQUENESS OF RLDA.

                                STARS
                                ;;*****************************************************************
                                ;TEST THE UNIQUENESS OF THE DISK ADDRESS REGISTER.  THE RLCS
                                ;AND RLBA ARE LOADED WITH XXX20X AND 177776 RESPECTIVELY.
                                ;THE RLDA IS THEN WRITTEN TO INSURE THAT
                                ;NEITHER THE RLCS OR THE RLBA ARE MODIFIED.
                                ; IF RLV12, CHECK THAT RLBAE IS UNAFFECTED ALSO.
                                STARS
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                        T17::
015036 012737 000200 003432             MOV     #CRDY,GDDAT     ;CONTROLLER READY
015044 032777 040000 166252             BIT     #DERR,@RLCS     ;IF DRIVE ERROR IS SET...
015052 001403                           BEQ     1$
015054 052737 140000 003432             BIS     #ERR!DERR,GDDAT ;...EXPECT IT BACK.
015062 013777 003432 166234 1$:         MOV     GDDAT,@RLCS     ;LOAD CS
015070 012777 177776 166230             MOV     #-2,@RLBA       ;LOAD BA WITH ALL 1'S
015076 023727 003402 000003             CMP     RLTYP,#RLV12X
015104 001003                           BNE     2$
015106 012777 000074 166220             MOV     #74,@RLBAE      ; LOAD BAE.
015114 005077 166210 2$:                CLR     @RLDA           ; WRITE TO RLDA.
015120 017737 166200 003434             MOV     @RLCS,BDDAT     ;READ RLCS
015126 042737 000001 003434             BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
015134 023737 003432 003434             CMP     GDDAT,BDDAT
015142 001405                           BEQ     3$              ; BR IF CS UNCHANGED.
015144                                  DFERR   EM76,ERR2       ;DA MODIFIED CS
015144 104455                                                                   TRAP    C$ERDF
015146 000174                                                                   .WORD   124
015150 033737                                                                   .WORD   EM76
015152 026220                                                                   .WORD   ERR2
015154                                  CKLOOP
015154 104406                                                                   TRAP    C$CLP1
015156 012737 177776 003432 3$:         MOV     #-2,GDDAT       ; SET EXPECTED BA.
015164 017737 166136 003434             MOV     @RLBA,BDDAT     ; READ IT.
015172 023737 003432 003434             CMP     GDDAT,BDDAT
015200 001405                           BEQ     4$              ; BR IF BA UNCHANGED.
015202                                  DFERR   EM77,ERR2       ;DA MODIFIED BA
015202 104455                                                                   TRAP    C$ERDF
015204 000175                                                                   .WORD   125
015206 033773                                                                   .WORD   EM77
015210 026220                                                                   .WORD   ERR2
015212                                  CKLOOP
015212 104406                                                                   TRAP    C$CLP1
015214 023727 003402 000003 4$:         CMP     RLTYP,#RLV12X
015222 001016                           BNE     5$
015224 012737 000074 003432             MOV     #74,GDDAT       ; SET EXPECTED BAE.
015232 017737 166076 003434             MOV     @RLBAE,BDDAT    ; READ IT.
015240 023737 003432 003434             CMP     GDDAT,BDDAT
015246 001404                           BEQ     5$              ; BR IF BAE UNCHANGED.
015250                                  DFERR   EM77A,ERR2      ; DA MODIFIED BAE.
015250 104455                                                                   TRAP    C$ERDF
015252 000176                                                                   .WORD   126
015254 034025                                                                   .WORD   EM77A
015256 026220                                                                   .WORD   ERR2
```

```
                                                                    E 6
 1660  015260                    5$:     ENDTST
   (3)  015260
   (3)  015260  104401                                    L10033:
                                                                TRAP    C$ETST
```

```
                                .SBTTL   18 -- UNIQUENESS OF RLMP.

                                STARS
                                ;;*****************************************************************
                                ;TEST THE UNIQUENESS OF THE MULTI-PURPOSE REGISTER
                                ;WE WILL WRITE THE RLCS, RLBA, AND THE RLDA, THEN THE
                                ;RLMP IS WRITTEN. WE THEN GO BACK AN VERIFY THE CONTENTS
                                ;OF THE RLCS, RLBA, RLDA.
                                ; IF RLV12, INCLUDE THE RLBAE ALSO.
                                STARS
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                        T18::
015262  012737  000200  003432          MOV     #CRDY,GDDAT     ;CONTROLLER READY
015270  032777  040000  166026          BIT     #DERR,@RLCS     ;IF DRIVE ERROR IS SET...
015276  001403                          BEQ     1$
015300  052737  140000  003432          BIS     #ERR!DERR,GDDAT ;...EXPECT IT BACK LATER.
015306  013777  003432  166010  1$:     MOV     GDDAT,@RLCS     ;LOAD CS
015314  012777  177776  166004          MOV     #-2,@RLBA       ;LOAD BA WITH ALL 1'S
015322  012777  177777  166000          MOV     #-1,@RLDA       ;LOAD RLDA
015330  023727  003402  000003          CMP     RLTYP,#RLV12X
015336  001003                          BNE     2$
015340  012777  000074  165766          MOV     #74,@RLBAE      ;LOAD RLBAE.
015346  005077  165760          2$:     CLR     @RLMP           ;WRITE TO RLMP
015352  017737  165746  003434          MOV     @RLCS,BDDAT     ;READ RLCS
015360  042737  000001  003434          BIC     #DRDY,BDDAT     ;IGNORE DRIVE READY
015366  023737  003432  003434          CMP     GDDAT,BDDAT
015374  001405                          BEQ     3$              ; BR IF CS OK.
015376                                  DFERR   EM44,ERR2       ;MP MODIFIED CS
015376  104455                                                          TRAP    C$ERDF
015400  000177                                                          .WORD   127
015402  032333                                                          .WORD   EM44
015404  026220                                                          .WORD   ERR2
015406                                  CKLOOP
015406  104406                                                          TRAP    C$CLP1
015410  012737  177776  003432  3$:     MOV     #-2,GDDAT       ; SET EXPECTED BA.
015416  017737  165704  003434          MOV     @RLBA,BDDAT     ; READ IT.
015424  023737  003432  003434          CMP     GDDAT,BDDAT
015432  001405                          BEQ     4$              ; BR IF BA OK.
015434                                  DFERR   EM45,ERR2       ;MP MODIFIED BA
015434  104455                                                          TRAP    C$ERDF
015436  000200                                                          .WORD   128
015440  032366                                                          .WORD   EM45
015442  026220                                                          .WORD   ERR2
015444                                  CKLOOP
015444  104406                                                          TRAP    C$CLP1
015446  012737  177777  003432  4$:     MOV     #-1,GDDAT       ; SET EXPECTED DA.
015454  017737  165650  003434          MOV     @RLDA,BDDAT     ; READ IT.
015462  023737  003432  003434          CMP     GDDAT,BDDAT
015470  001405                          BEQ     5$              ; BR IF DA OK.
015472                                  DFERR   EM46,ERR2       ;MP MODIFIED DA
015472  104455                                                          TRAP    C$ERDF
015474  000201                                                          .WORD   129
015476  032421                                                          .WORD   EM46
015500  026220                                                          .WORD   ERR2
015502                                  CKLOOP
```

```
                                                                                                                    G 6
     (3)  015502  104406                                                                                                    TRAP    C$CLP1
1701      015504  023727  003402  000003  5$:     CMP     RLTYP,#RLV12X
1702      015512  001016                          BNE     6$
1703      015514  012737  000074  003432          MOV     #74,GDDAT       ; SET EXPECTED BAE.
1704      015522  017737  165606  003434          MOV     @RLBAE,BDDAT    ; READ IT.
1705      015530  023737  003432  003434          CMP     GDDAT,BDDAT
1706      015536  001404                          BEQ     6$              ; BR IF BAE OK.
1707      015540                                  DFERR   EM47,ERR2       ; MP MODIFIED BAE.
     (5)  015540  104455                                                                                                    TRAP    C$ERDF
     (6)  015542  000202                                                                                                    .WORD   130
     (6)  015544  032454                                                                                                    .WORD   EM47
     (6)  015546  026220                                                                                                    .WORD   ERR2
1708      015550                          6$:     ENDTST
     (3)  015550                                                                                        L10034:
     (3)  015550  104401                                                                                                    TRAP    C$ETST
```

```
                                                                    M 6
1710                                    .SBTTL   19 -- UNIQUENESS OF RLBAE (RLV12 ONLY).
1711
1712  015552                            STARS
  (2)                                   ;;*****************************************************************
1713                                    ;TEST THAT WRITING TO THE RLBAE HAS NO AFFECT ON
1714                                    ;THE RLBA AND RLDA REGISTERS.  THE RLCS REGISTER WILL
1715                                    ;BE AFFECTED, BUT ONLY IN THE EXTENDED ADDRESS BITS <5:4>,
1716                                    ;WHICH SHOULD MIRROR RLBAE<1:0>.
1717  015552                            STARS
  (2)                                   ;;*****************************************************************
1718  015552                                    BEGIN.TEST
  (4)  015552                                                                       T19::
1719  015552  023727  003402  000003            CMP     RLTYP,#RLV12X   ; RLV12 WITH BAE ??
1720  015560  001071                           BNE     4$              ; EXIT IF NOT.
1721  015562  012737  000200  003432            MOV     #CRDY,GDDAT     ;CONTROLLER READY
1722  015570  032777  040000  165526            BIT     #DERR,@RLCS     ;IF DRIVE ERROR IS SET...
1723  015576  001403                           BEQ     1$
1724  015600  052737  140000  003432            BIS     #ERR!DERR,GDDAT ;...EXPECT IT BACK LATER.
1725  015606  013777  003432  165510  1$:       MOV     GDDAT,@RLCS     ;LOAD CS, NOTE THAT <5:4> = 0
1726  015614  005077  165506                    CLR     @RLBA           ; 0 => BA
1727  015620  005077  165504                    CLR     @RLDA           ; 0 => DA
1728  015624  012777  000077  165502            MOV     #77,@RLBAE      ; WRITE TO RLBAE.
1729  015632  052737  000060  003432            BIS     #60,GDDAT       ; SET EXPECTED CS.
1730  015640  017737  165460  003434            MOV     @RLCS,BDDAT     ; READ IT.
1731  015646  042737  000001  003434            BIC     #DRDY,BDDAT     ; IGNORE DRIVE READY BIT.
1732  015654  023737  003432  003434            CMP     GDDAT,BDDAT
1733  015662  001405                           BEQ     2$              ; BR IF CS IS RIGHT.
1734  015664                                    DFERR   EM50,ERR2       ; CS WRONG AFTER WRITING BAE.
  (5)  015664  104455                                                               TRAP    C$ERDF
  (6)  015666  000203                                                               .WORD   131
  (6)  015670  032510                                                               .WORD   EM50
  (6)  015672  026220                                                               .WORD   ERR2
1735  015674                                    CKLOOP
  (3)  015674  104406                                                               TRAP    C$CLP1
1736  015676  005037  003432        2$:         CLR     GDDAT           ; EXPECT ZERO ON THE REST.
1737  015702  017737  165420  003434            MOV     @RLBA,BDDAT     ; READ BA
1738  015710  001405                           BEQ     3$              ; BR IF UNAFFECTED.
1739  015712                                    DFERR   EM51,ERR2       ; BAE MODIFIED BA.
  (5)  015712  104455                                                               TRAP    C$ERDF
  (6)  015714  000204                                                               .WORD   132
  (6)  015716  032547                                                               .WORD   EM51
  (6)  015720  026220                                                               .WORD   ERR2
1740  015722                                    CKLOOP
  (3)  015722  104406                                                               TRAP    C$CLP1
1741  015724  017737  165400  003434  3$:       MOV     @RLDA,BDDAT     ; READ DA.
1742  015732  001404                           BEQ     4$              ; BR IF UNAFFECTED.
1743  015734                                    DFERR   EM52,ERR2       ; BAE MODIFIED DA.
  (5)  015734  104455                                                               TRAP    C$ERDF
  (6)  015736  000205                                                               .WORD   133
  (6)  015740  032573                                                               .WORD   EM52
  (6)  015742  026220                                                               .WORD   ERR2
1744  015744                            4$:     ENDTST
  (3)  015744                                                               L10035:
  (3)  015744  104401                                                               TRAP    C$ETST
```

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

                                STARS
                                ;;*****************************************************************
                                ; TEST FUNCTION CODE 0.
                                ; IF RL11 -- NOP.  EXPECT CS<13:10> CLEAR, OTHER REGISTERS UNAFFECTED.
                                ; IF RLVXX -- MAINT.  SETUP IN NORMAL FASHION, BUT ONLY CHECK FOR
                                ;               FUNCTION COMPLETE (NO ERRORS AND DA = INITIAL DA+6).
                                ;               WE'LL CHECK THE REST OF THE MAINT FUNCTION LATER.
                                STARS
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                                T20::
                                ;
                                ; INITIAL SETUP ASSUMES THAT WE HAVE AN RLV.
                                ; IF NOT -- NOT TO WORRY -- IT SHOULD JUST NO-OP.
                                ; ALL WE CARE ABOUT IS THAT WE GET NO ERRORS (OR HUNG RL).
                                ;
016746 004537 010714            1$:     JSR     R5,CALCRC       ; MAKE A CRC EVEN THO WE WON'T CHECK IT.
016752 000000                           .WORD   0               ; THIS WILL BE THE INITIAL DA WORD.
016754 004537 010210                    JSR     R5,LDFUN        ; LOAD UP THE FUNCTION.
016760 000000 177001                    0, -511.                ; FUNCTION 0, WC -511.
016764 004537 010640                    JSR     R5,WTCRDY       ; WAIT FOR CONTROLLER.
016770 023727 003402 000000             CMP     RLTYP,#RL11     ; NOW ARE WE RL11 ??
016776 001047                           BNE     6$              ;CHECK 'MAINT" FINAL STATE IF NOT.
                                ;
                                ; RL11 -- CHECK FOR ERRORS AFTER THE NO-OP.
                                ;
016000 005037 003356            2$:     CLR     B.MP            ; MPR WILL BE 0.
016004 013737 003350 003432             MOV     B.CS,GDDAT      ; EXPECT "NO-OP" SHOULD HAVE...
016012 042737 036000 003432             BIC     #036000,GDDAT   ;...CLEARED ANY ERROR BITS <13:10>...
016020 013737 003364 003434             MOV     E.CS,BDDAT      ;...THAT MIGHT HAVE BEEN ON.
016026 023737 003432 003434             CMP     GDDAT,BDDAT     ; IS THAT WHAT WE HAVE ??
016034 001405                           BEQ     3$              ; YES.
016036                                  DFERR   EM14,ERR2       ; CS WRONG AFTER NOP.
016036 104455                                                           TRAP    C$ERDF
016040 000206                                                           .WORD   134
016042 031405                                                           .WORD   EM14
016044 026220                                                           .WORD   ERR2
016046                                  CKLOOP
016046 104406                                                           TRAP    C$CLP1
016050 023737 003366 003352     3$:     CMP     E.BA,B.BA       ; BA UNALTERED ??
016056 001010                           BNE     4$
016060 023737 003370 003354             CMP     E.DA,B.DA       ; DA UNALTERED ??
016066 001004                           BNE     4$
016070 023737 003372 003356             CMP     E.MP,B.MP       ; MP UNALTERED ??
016076 001405                           BEQ     5$              ; EXIT IF ALL THREE WERE OK.
016100                          4$:     DFERR   EM14A,ERR0      ; REGISTERS ALTERED BY NOP.
016100 104455                                                           TRAP    C$ERDF
016102 000207                                                           .WORD   135
016104 031440                                                           .WORD   EM14A
016106 026164                                                           .WORD   ERR0
016110                                  CKLOOP
016110 104406                                                           TRAP    C$CLP1
016112                          5$:     EXIT    TST             ; RL NOP TEST DONE.
016112 104432                                                           TRAP    C$EXIT
```

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

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

016160                          STARS
  (2)                           ;;*****************************************************************
                                ;TEST THAT THE RL CAN INTERRUPT THE CPU.
                                ;WE'LL SET CPU PRIORITY AT 0, AND EXECUTE NOP/MAINT, AS BEFORE.
                                ;THE INTERRUPT SERVICE DOES NOTHING BUT SET A FLAG.
                                ;WE'LL WAIT 800 MSEC FOR THAT FLAG, BEFORE DECLARING AN ERROR.
                                ;WRONG VECTORS UNDER 1000 ARE TRAPPED BY THE DRS.
                                ;WRONG VECTORS ABOVE 1000 ARE BIG TROUBLE !!!!
016160                          STARS
  (2)                           ;;*****************************************************************
016160                                  BEGIN.TEST
  (4)   016160                                                                  T21::
016160  005037  010634                  CLR     INTFLG          ;CLEAR INTERRUPT FLAG
016164                                  SETPRI  #PRI00          ;SET PSW TO 0
  (3)   016164  012700  000000                                                  MOV     #PRI00,R0
  (3)   016170  104441                                                          TRAP    C$SPRI
016172  004537  010714                  JSR     R5,CALCRC       ; AGAIN, CRC WON'T BE CHECKED.
016176  000000                          .WORD   0
016200  004537  010210                  JSR     R5,LDFUN        ; LOAD UP AND EXECUTE...
016204  000100  177001                  INTEN!0, -511.          ; NOP/MAINT AND INTERRUPT.
016210  004537  010640                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY.
016214  005737  010634                  TST     INTFLG          ;DID INTERRUPT OCCUR ??
016220  001006                          BNE     1$              ; BR IF SO.
016222  005037  003424                  CLR     TMP0            ; CPU LEVEL FOR ERROR MSG.
016226                                  DFERR   EM15,ERR7       ; INTERRUPT NOT REC'D.
  (5)   016226  104455                                                          TRAP    C$ERDF
  (6)   016230  000211                                                          .WORD   137
  (6)   016232  031544                                                          .WORD   EM15
  (6)   016234  026370                                                          .WORD   ERR7
016236                          1$:     SETPRI  #PRI07
  (3)   016236  012700  000340                                                  MOV     #PRI07,R0
  (3)   016242  104441                                                          TRAP    C$SPRI
016244                                  ENDTST
  (3)   016244                                                          L10037:
  (3)   016244  104401                                                          TRAP    C$ETST
```

```
                                                                L 6
1825                                    .SBTTL   22 -- TEST INTERRUPT PRIORITY BR LEVEL.
1826
1827  016246                            STARS
  (2)                                   ;;*****************************************************************
1828                                    ;TEST THAT PRIORITY GIVEN IS ACTUAL PRIORITY OF CONTROLLER.
1829                                    ;WE ALREADY KNOW THAT THE CONTROLLER CAN INTERRUPT.  WE'LL START
1830                                    ;AT CPU LEVEL 7 AND WORK DOWN 'TIL THE INTERRUPT COMES IN.
1831                                    ; AS BEFORE, WE'RE USING THE NOP/MAINT FUNCTION.
1832                                    ; NOTE THAT RLV11 IS FIXED AT BR4, RL11 AND RLV12 ARE PROGRAMABLE
1833                                    ; AND DEFAULT TO BR5.  LOC 'BPRIOR' IS PROPERLY SET AT INIT TIME.
1834  016246                            STARS
  (2)                                   ;;*****************************************************************
1835  016246                                  BEGIN.TEST
  (4)  016246                                                                   T22::
1836  016246  012737  000340  003434          MOV     #PRI07,BDDAT    ;SET UP INITIAL OF 7
1837  016254  012737  000007  003424          MOV     #7,TMP0         ; A COPY FOR ERROR MSG.
1838  016262  013737  003340  003432          MOV     BPRIOR,GDDAT    ;GET GIVEN PRIORITY
1839  016270                                  BGNSEG                  ; **** START SEGMENT ****
  (3)  016270  104404                                                           TRAP    C$BSEG
1840  016272                          1$:     SETPRI  BDDAT           ; SET CURRENT PRIORITY...
  (3)  016272  013700  003434                                                   MOV     BDDAT,R0
  (3)  016276  104441                                                           TRAP    C$SPRI
1841  016300  005037  010634                  CLR     INTFLG          ;...AND CLEAR INTERRUPT FLAG.
1842  016304  004537  010714                  JSR     R5,CALCRC       ; JUST LIKE BEFORE.
1843  016310  000000                          .WORD   0
1844  016312  004537  010210                  JSR     R5,LDFUN        ; LOAD AND EXECUTE...
1845  016316  000100  177001                  INTEN!0, -511.          ;...NOP/MAINT AND INTERRUPT.
1846  016322  004537  010640                  JSR     R5,WTCRDY       ;WAIT FOR CONTROLLER READY.
1847  016326  023737  003434  003432          CMP     BDDAT,GDDAT     ;SHOULD IT INTERRUPT
1848  016334  002004                          BGE     2$              ; BR IF NOT.
1849  016336  005737  010634                  TST     INTFLG          ; DID IT INTERRUPT ??
1850  016342  001011                          BNE     4$              ; PROCEED IF SO.
1851  016344  000403                          BR      3$              ; NO, ERROR.
1852  016346  005737  010634          2$:     TST     INTFLG          ; DID IT INTERRUPT ??
1853  016352  001405                          BEQ     4$              ; PROCEED IF NOT.
1854  016354                          3$:     DFERR   EM16,ERR7       ; INTERRUPT PRIORITY FAILURE.
  (5)  016354  104455                                                           TRAP    C$ERDF
  (6)  016356  000212                                                           .WORD   138
  (6)  016360  031606                                                           .WORD   EM16
  (6)  016362  026370                                                           .WORD   ERR7
1855  016364                                  CKLOOP
  (3)  016364  104406                                                           TRAP    C$CLP1
1856  016366  162737  000040  003434  4$:     SUB     #40,BDDAT       ;LOWER CPU PRIORITY...
1857  016374  005337  003424                  DEC     TMP0            ;...TO NEXT LEVEL.
1858  016400  100334                          BPL     1$
1859  016402                                  ENDSEG
  (3)  016402                                                           10000$:
  (3)  016402  104405                                                           TRAP    C$ESEG
1860  016404                                  SETPRI  #PRI07
  (3)  016404  012700  000340                                                   MOV     #PRI07,R0
  (3)  016410  104441                                                           TRAP    C$SPRI
1861  016412                                  ENDTST
  (3)  016412                                                           L10040:
  (3)  016412  104401                                                           TRAP    C$ETST
```

```
                                                        M 6
 1863                                   .SBTTL   23 -- RLV11/12 MAINTENANCE, FORCED OPI (WC <> 511.)
 1864
 1865  016414                           STARS
   (2)                                  ;;*****************************************************************************
 1866                                   ;EXECUTE MAINTENANCE MODE WITH AN INCORRECT INITIAL WORD COUNT.
 1867                                   ;FIRST PASS WC < 510. AND SECOND PASS WC > 511.  BOTH CASES SHOULD
 1868                                   ;FORCE OPI ERRORS. THE TEST SHOULD FORCE COMPOSITE ERROR(BIT 15),
 1869                                   ;HEADER NOT FOUND(BIT12) AND OPI(BIT 10). DRIVE ERROR IS IGNORED.
 1870  016414                           STARS
   (2)                                  ;;*****************************************************************************
 1871  016414                                 BEGIN.TEST
   (4)  016414                                                                          T23::
 1872                                   ;*
 1873                                   ;* NOTE: IF CONTROLLER TYPE IS RL11, YOU HAVE END-PASS AT THIS POINT.
 1874                                   ;* ALL REMAINING TESTS ARE UNIQUE TO RLV11 AND/OR RLV12.
 1875                                   ;*
 1876  016414  004537  010714          1$:    JSR     R5,CALCRC       ;DO CRC CALCULATION FOR...
 1877  016420  000000                         .WORD   0               ;...INITIAL DA OF ZERO.
 1878  016422  012737  177003  016452        MOV     #-509.,3$       ; 1ST WC < 510.
 1879  016430  112737  000074  032251        MOVB    #'<,EM27X       ; ADJUST ERROR TEXT.
 1880  016436  112737  000060  032255        MOVB    #'0,EM27X+4
 1881  016444  004537  010210          2$:    JSR     R5,LDFUN        ;PERFORM MAINT FUNCTION
 1882  016450  000000                         MAINT
 1883  016452  177003                  3$:    -509.                   ; TEST WORD COUNT.
 1884  016454  004537  010640                JSR     R5,WTCRDY
 1885  016460  012700  112000                MOV     #ERR!HNF!OPI,R0 ;EXPECT AT LEAST THESE ERRORS.
 1886  016464  043700  003364                BIC     E.CS,R0
 1887  016470  001410                        BEQ     4$              ; EXIT IF WE'RE OK.
 1888  016472  004537  007510                JSR     R5,GETERR       ; OTHERWISE, GET THEM ALL...
 1889  016476  000240                        NOP                     ;...NONE IS EQUALLY BAD NEWS.
 1890  016500                               DFERR   EM27,ERR10      ; STATUS INCORRECT, WC <> 511.
   (5)  016500  104455                                                                  TRAP    C$ERDF
   (6)  016502  000213                                                                  .WORD   139
   (6)  016504  032151                                                                  .WORD   EM27
   (6)  016506  026420                                                                  .WORD   ERR10
 1891  016510                               CKLOOP
   (3)  016510  104406                                                                  TRAP    C$CLP1
 1892  016512  023727  016452  177000  4$:    CMP     3$,#-512.       ; 2ND PASS ??
 1893  016520  001412                        BEQ     5$              ; WE'RE DONE IF SO.
 1894  016522  012737  177000  016452        MOV     #-512.,3$       ; OTHERWISE, SET WC > 511.
 1895  016530  112737  000076  032251        MOVB    #'>,EM27X       ; ADJUST ERROR TEXT.
 1896  016536  112737  000061  032255        MOVB    #'1,EM27X+4
 1897  016544  000737                        BR      2$              ; AND GO 'ROUND.
 1898  016546                          5$:    ENDTST
   (3)  016546                                                                  L10041:
   (3)  016546  104401                                                                  TRAP    C$ETST
```

```
                                                    N 6
1901                                .SBTTL   24 -- RLV11/12 MAINTENANCE, FORCED OPI INTERRUPT.
1902
1903  016550                        STARS
  (2)
1904                                ;;*************************************************************************
1905                                ;PERFORM TEST OF INTERRUPT BY ISSUING RLV11 MAINTENANCE FUNCTION
1906                                ;WITH A WORD COUNT OF -509. TO FORCE AN OPI ERROR.
1907  016550                        ;CHECK THAT INTERRUPT OCCURS, REPORT ERROR IF NOT.
  (2)                               STARS
1908  016550                        ;;*************************************************************************
  (4)  016550                             BEGIN.TEST
1909  016550  004537  010714                                                          T24::
1910  016554  000000               1$:     JSR     R5,CALCRC       ;CALCULATE CRC FOR...
1911  016556                                .WORD   0               ;...INITIAL DA OF ZERO.
  (3)  016556  012700  000000              SETPRI  #PRI00          ;SET PRIORITY TO ZERO
  (3)  016562  104441                                                               MOV     #PRI00,R0
1912  016564  005037  010634                                                       TRAP    C$SPRI
1913  016570  004537  010210               CLR     INTFLG          ;CLEAR INT. FLAG
1914  016574  000100  177003               JSR     R5,LDFUN
1915  016600  004537  010640               MAINT!INTEN, -509.      ; INVALID WC TO FORCE OPI ERROR.
1916  016604  005737  010634               JSR     R5,WTCRDY       ;WAIT FOR READY
1917  016610  001005                       TST     INTFLG          ; DID IT INTERRUPT ??
1918  016612                               BNE     4$              ; YES, THAT'S ALL.
  (5)  016612  104455                       DFERR   EM17,ERR0       ; NO INTERRUPT RECEIVED.
  (6)  016614  000214                                                               TRAP    C$ERDF
  (6)  016616  031641                                                               .WORD   140
  (6)  016620  026164                                                               .WORD   EM17
1919  016622                                                                        .WORD   ERR0
                                    CKLOOP
  (3)  016622  104406                                                               TRAP    C$CLP1
1920  016624
  (3)  016624  012700  000340       4$:     SETPRI  #PRI07
  (3)  016630  104441                                                               MOV     #PRI07,R0
1921  016632                                                                        TRAP    C$SPRI
  (3)  016632
  (3)  016632  104401               ENDTST
                                                                    L10042:
                                                                                  TRAP    C$ETST
```

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

016634                          STARS
  (2)                           ;;*****************************************************************
                                ;NOTE: THE SOFT TIMER EMPLOYED HERE IS CALIBRATED FOR LSI'S ONLY.
                                ;      IF THE CPU TYPE IS UNKNOWN (NOT LSI), BYPASS THIS TEST.
                                ;
                                ;PERFORM RLV11 MAINTENANCE FUNCTION (0) IN FLAG MODE.
                                ;FORCE AN OPI TIMEOUT BY SETTING AN INVALID WORD COUNT.
                                ;MEASURE THE TIME UNTIL THE ERROR FLAG SETS AND COMPARE THAT
                                ;TIME AGAINST THE SPEC LIMITS (155 TO 650 MSEC).
                                ; NOTE: SINCE THE TIMING LOOP IS SO GROSS, WE'LL GIVE IT
                                ; A SECOND CHANCE BEFORE WE DECLARE AN ERROR.
016634                          STARS
  (2)                           ;;*****************************************************************
016634                                BEGIN.TEST
  (4)   016634                                                                          T25::
016634  005737  003400                  TST     CPUTYP          ; LSI CPU ???
016640  100447                          BMI     5$              ; IF NOT, DON'T EVEN TRY !!!
016642  004537  010714          1$:     JSR     R5,CALCRC       ; MAKE CRC FOR ZERO.
016646  000000                          .WORD   0
016650  012704  000002                  MOV     #2,R4
016654  004537  010210          2$:     JSR     R5,LDFUN        ;PERFORM MAINT. FUNCTION
016660  000000  177003                  MAINT, -509.            ; MAINT WITH INVALID WC.
016664  004537  010640                  JSR     R5,WTCRDY       ; WAIT FOR DONE.
016670  012737  001440  003434          MOV     #800.,BDDAT     ; CALCULATE OPI TIME...
016676  166637  177774  003434          SUB     -4(SP),BDDAT    ;...IN BDDAT.
016704  032777  002000  164412          BIT     #OPI,@RLCS      ; OPI ERROR ??
016712  001005                          BNE     3$              ; YES, CHECK TIMING.
016714                                  DFERR   EM31,ERR0       ; OPI FLAG NOT RECEIVED.
  (5)   016714  104455                                                          TRAP    C$ERDF
  (6)   016716  000215                                                          .WORD   141
  (6)   016720  032260                                                          .WORD   EM31
  (6)   016722  026164                                                          .WORD   ERR0
016724  000415                          BR      5$
016726  023737  003434  003450  3$:     CMP     BDDAT,OPIMX
016734  003004                          BGT     4$
016736  023737  003434  003446          CMP     BDDAT,OPIMN
016744  002005                          BGE     5$              ; OK IF TIME WITHIN LIMITS.
016746  077436                  4$:     SOB     R4,2$           ; TRY ONCE MORE.
016750                                  DFERR   EM32,ERR11      ;OPI TIMING INCORRECT
  (5)   016750  104455                                                          TRAP    C$ERDF
  (6)   016752  000216                                                          .WORD   142
  (6)   016754  032306                                                          .WORD   EM32
  (6)   016756  026450                                                          .WORD   ERR11
016760                          5$:     ENDTST
  (3)   016760                                                          L10043:
  (3)   016760  104401                                                          TRAP    C$ETST
```

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

                                STARS
                                ;;*****************************************************************
                                ;PERFORM RLV11/12 MAINTENANCE FUNCTION 0 IN FLAG MODE AND CHECK
                                ;FOR PROPER INCREMENT OF THE DA AND BA REGISTERS. CHECK THE SERIAL
                                ;WRITE/READ DATA PATHS BY READING OUT OF THE FIFO VIA THE MP REGISTER
                                ;THE CRC OF DA+3 AND THE CRC OF CRC OF DA+4 AND COMPARING WITH EXPECTED
                                ;RESULTS. CHECK THE TRANSFER OF 255 WORDS FROM BUF1 MEMORY THROUGH THE
                                ;FIFO INTO BUF2 MEMORY FOR PROPER DATA.
                                ;CHECK THE PREVIOUSLY WRITTEN DATA IN THE LAST WORD+1 OF BUF2 FOR
                                ;A VALUE:123456 TO INSURE THAT THE TRANSFER WAS NOT MORE THAN 255 WORDS.
                                ;DMA 1 -- USES 28 DATA PATTERNS IN "PATCRC" AND "PATDAT".
                                ;DMA 2 -- USES COMPLIMENT OF THOSE DATA PATTERNS.
                                ;DMA 3 -- USES A RANDOM 256 WORD PATTERN.
                                STARS
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                        T26::
                                ;
                                ; SUBTEST 1 -- 28 DIFFERENT DATA PATTERNS.
                                ;
                                DMA1:   BGNSUB
                                                                                        T26.1:
                                                                                                TRAP    C$BSUB
                                        MOV     #PATCRC,R4      ;SET TABLE POINTER.
017002                          1$:     MOV     72(R4),2$       ; GET TEST DATA...
                                        JSR     R5,SETPAT       ;...AND FILL THE BUFFER.
                                2$:     .WORD   0
                                        MOV     (R4)+,3$        ; GET DA TEST WORD...
                                        JSR     R5,CALCRC       ;...AND CALCULATE CRC'S.
                                3$:     .WORD   0
017132                                  JSR     PC,EXECUT       ; XCT MAINT MODE AND CHECK RESULTS.
                                        TST     (R4)            ; LAST PATTERN DONE ??
                                        BNE     1$              ; LOOP IF NOT.
                                        ENDSUB
                                                                                        L10045:
                                                                                                TRAP    C$ESUB
                                ;
                                ; SUBTEST 2 -- THE COMPLIMENT OF THE ABOVE 28 PATTERNS.
                                ;
                                DMA2:   BGNSUB
                                                                                        T26.2:
                                                                                                TRAP    C$BSUB
                                        MOV     #PATCRC,R4      ;RESET TABLE POINTER.
017050                          1$:     MOV     72(R4),2$
                                        JSR     R5,SETCMP       ; FILL WITH COMPLIMENT DATA.
                                2$:     .WORD   0
                                        MOV     (R4)+,3$
                                        JSR     R5,CALCRC       ; CALCULATE CRC'S.
                                3$:     .WORD   0
017132                                  JSR     PC,EXECUT       ; XCT
                                        TST     (R4)            ; LAST ONE DONE ??
                                        BNE     1$              ; LOOP IF NOT.
                                        ENDSUB
                                                                                        L10046:
```

```
                                                                                                    TRAP    C$ESUB
                        ;
                        ; SUBTEST 3 -- ONE RANDOM DATA PATTERN.
                        ;
                        DMA3:   BGNSUB
                                                                                                    T26.3:
                                                                                                    TRAP    C$BSUB
                                JSR     R5,SETRAN       ;FILL BUFFER WITH RANDOM DATA.
                                MOV     TEMLO,1$        ; USE TEMLO FOR THE CRC WORD.
                                JSR     R5,CALCRC       ; CALCULATE CRC'S.
                        1$:     .WORD   0
                                JSR     PC,EXECUT       ; XCT
                                ENDSUB
                                                                                                    L10047:
                                                                                                    TRAP    C$ESUB
                                EXIT    TST             ; ALL DONE.
                                                                                                    TRAP    C$EXIT
                                                                                                    .WORD   L10044-.
                        ;
                        ; THIS SUBROUTINE EXECUTES THE FUNCTION AND VERIFIES RESULTS.
                        ;
                        EXECUT: BGNSEG                  ; *** SET LOOP SEGMENT ***
                                                                                                    TRAP    C$BSEG
                                JSR     R5,LDFUN        ;PERFORM MAINT. FUNCTION
                                MAINT, -511.
                                JSR     R5,WTCRDY       ;WAIT FOR READY
                                JSR     R5,GETERR       ;CHECK CONTROLLER FOR ERRORS
                                BR      11$             ; BR IF NONE.
                                DFERR   EM98,ERR6       ;UNEXPECTED CONTROLLER ERRORS.
                                                                                                    TRAP    C$ERDF
                                                                                                    .WORD   143
                                                                                                    .WORD   EM98
                                                                                                    .WORD   ERR6
                                CKLOOP
                                                                                                    TRAP    C$CLP1
                        11$:    BIT     #OPI!NXM,E.CS   ; OPI OR NXM ??
                                BEQ     12$             ; PROCEED IF NOT.
                                EXIT    SEG             ; YES, DON'T BOTHER WITH ANY MORE.
                                                                                                    TRAP    C$EXIT
                                                                                                    .WORD   10000$-.
                        12$:    MOV     #BUFEND-2,GDDAT
                                MOV     E.BA,BDDAT
                                CMP     GDDAT,BDDAT     ;FINAL BA CORRECT ??
                                BEQ     1$
                                DFERR   EM10,ERR4       ;FINAL BA INCORRECT.
                                                                                                    TRAP    C$ERDF
                                                                                                    .WORD   144
                                                                                                    .WORD   EM10
                                                                                                    .WORD   ERR4
                                CKLOOP
                                                                                                    TRAP    C$CLP1
                        1$:     MOV     B.DA,GDDAT      ;GET BEFORE DA REGISTER
                                MOV     E.DA,BDDAT
                                CLR     TMP0
                                MOVB    B.DA,TMP0
                                ADD     #6,TMP0         ;+6 TO DA LOW BYTE
```

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

```
                                                                                        MOV     #5,-(SP)
                                                                                        MOV     SP,R0
                                                                                        TRAP    C$PNTX
                                                                                        ADD     #14,SP
                        CKLOOP
                                                                                        TRAP    C$CLP1
                7$:     TST     (R2)+           ;INCREMENT BUFFER POINTERS.
                        TST     (R3)+
                        INC     R1              ;AND THE WORD COUNT.
                        CMP     R1,#255.        ; DONE ALL WORDS ??
                        BLT     5$              ; NOT YET, CONTINUE.
                        TST     TMP2            ; YES, ANY ERRORS LOGGED ??
                        BEQ     8$              ;NO.
                        PRINTB  #FRMT15,TMP2    ;YES, PRINT SUMMARY.
                                                                                        MOV     TMP2,-(SP)
                                                                                        MOV     #FRMT15,-(SP)
                                                                                        MOV     #2,-(SP)
                                                                                        MOV     SP,R0
                                                                                        TRAP    C$PNTB
                                                                                        ADD     #6,SP
003432          8$:     MOV     #123456,GDDAT   ;EXPECTED DATA IN LAST WORD+1
                        MOV     (R2),BDDAT      ;GET LAST WORD+1 FROM BUF2
003434                  CMP     GDDAT,BDDAT
                        BEQ     9$
                        DFERR   EM23,ERR4       ; LAST+1 INCORRECT.
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   149
                                                                                        .WORD   EM23
                                                                                        .WORD   ERR4
                9$:     ENDSEG
                                                                                10000$:
                                                                                        TRAP    C$ESEG
                        RTS     PC              ; RETURN.

                10$:    ENDTST
                                                                                L10044:
                                                                                        TRAP    C$ETST
```

```
                                .SBTTL   27 -- RLV11/12 MAINTENANCE, FIFO ADDRESSING.

                                STARS
                                ;;*****************************************************************
                                ;TEST THAT FIFO OPERATES CORRECTLY.  STORE ADDRESS PATTERN
                                ;IN BUF1 (0-255) THAT CONTAINS A UNIQUE PATTERN IN EACH LOCATION.
                                ;PERFORM MAINTENANCE FUNCTION AND TEST BUF2 FOR PROPER FIFO
                                ;ADDRESSING.  NOTE THAT CRC'S ARE NOT CHECKED IN THIS TEST.
                                ;REPEAT A SECOND TIME USING A COMPLIMENT ADDRESS PATTERN.
                                ; ALSO USE AND TEST FOR INTERRUPT ON MAINT DONE.
                                STARS
                                ;;*****************************************************************
                                        BEGIN.TEST
                                                                                        T27::
                                ;
                                ; SUBTEST 1 -- FIFO ADDRESS PATTERN.
                                ;
017662                          FIFO1:  BGNSUB
                                                                                        T27.1:
017662  104402                                                                            TRAP    C$BSUB
017664  005001                          CLR     R1
017666  012702  000400                  MOV     #256.,R2
017672  012703  004364                  MOV     #BUF1,R3        ;SETUP TO STORE PATTERN IN BUF1
017676  010123                  1$:     MOV     R1,(R3)+
017700  005201                          INC     R1              ;INC. PATTERN
017702  005302                          DEC     R2
017704  001374                          BNE     1$
017706  004737  017754                  JSR     PC,XFIFO        ; EXECUTE AND CHECK FIFO DATA.
017712                                  ENDSUB
                                                                                        L10051:
017712  104403                                                                          TRAP    C$ESUB
                                ;
                                ; SUBTEST 2 -- FIFO COMPLIMENT ADDRESS PATTERN.
                                ;
017714                          FIFO2:  BGNSUB
                                                                                        T27.2:
017714  104402                                                                          TRAP    C$BSUB
017716  012701  177777                  MOV     #-1,R1
017722  012702  000400                  MOV     #256.,R2
017726  012703  004364                  MOV     #BUF1,R3
017732  010123                  1$:     MOV     R1,(R3)+        ; STORE COMPLIMENT ADDRESS PATTERN.
017734  005301                          DEC     R1
017736  005302                          DEC     R2
017740  001374                          BNE     1$
017742  004737  017754                  JSR     PC,XFIFO        ; EXECUTE AND CHECK FIFO DATA.
017746                                  ENDSUB
                                                                                        L10052:
017746  104403                                                                          TRAP    C$ESUB
017750                                  EXIT    TST             ; ALL DONE.
017750  104432                                                                          TRAP    C$EXIT
017752  000332                                                                          .WORD   L10050-.
                                ;
                                ; THIS SUBROUTINE EXECUTES THE MAINT FUNCTION AND TESTS RESULTS.
                                ;
017754                          XFIFO:  BGNSEG                  ;*** LOOP SEGMENT ***
017754  104404                                                                          TRAP    C$BSEG
```

```
                                        ;SETUP TO CLEAR BUF2
002134  017756  012702  000400            MOV     #256.,R2
002135  017762  012703  005364            MOV     #BUF2,R3
002136  017766  005023            1$:     CLR     (R3)+
002137  017770  005302                    DEC     R2
002140  017772  001375                    BNE     1$
002141  017774  005037  010634            CLR     INTFLG          ;CLEAR INT. FLAG
002144  020000                            SETPRI  #PRI00
    (3) 020000  012700  000000                                              MOV     #PRI00,R0
    (3) 020004  104441                                                      TRAP    C$SPRI
002146  020006  004537  010210            JSR     R5,LDFUN        ;LOAD FUNCTION
002152  020012  000100  177001            MAINT!INTEN, -511.
002156  020016  004537  010640            JSR     R5,WTCRDY       ;WAIT FOR READY
002162  020022  004537  007510            JSR     R5,GETERR       ;CHECK CONTROLLER FOR ERRORS
002166  020026  000405                    BR      2$              ; BR IF NONE.
002170  020030                            DFERR   EM98,ERR6       ;UNEXPECTED CONTROLLER ERRORS.
    (5) 020030  104455                                                      TRAP    C$ERDF
    (6) 020032  000226                                                      .WORD   150
    (6) 020034  034347                                                      .WORD   EM98
    (6) 020036  026340                                                      .WORD   ERR6
002174  020040                            CKLOOP
    (3) 020040  104406                                                      TRAP    C$CLP1
002176  020042  032737  022000  003364  2$:     BIT     #OPI!NXM,E.CS   ; OPI OR NXM ??
002204  020050  001110                    BNE     8$              ; QUIT IF SO.
002206  020052  005737  010634            TST     INTFLG          ;CHECK FOR INTERRUPT
002212  020056  001005                    BNE     4$
002214  020060                            DFERR   EM15,ERR0       ; DIDN'T INTERRUPT.
    (5) 020060  104455                                                      TRAP    C$ERDF
    (6) 020062  000227                                                      .WORD   151
    (6) 020064  031544                                                      .WORD   EM15
    (6) 020066  026164                                                      .WORD   ERR0
002220  020070                            CKLOOP
    (3) 020070  104406                                                      TRAP    C$CLP1
002222  020072  005037  003430    4$:     CLR     TMP2            ;CLEAR BAD WORD COUNTER
002226  020076  012703  004364            MOV     #BUF1,R3        ;GOOD DATA STORED IN BUF1
002232  020102  012702  005364            MOV     #BUF2,R2        ;DATA BUFFER WRITTEN INTO BY MAINT.
002236  020106  005001                    CLR     R1              ; WORD COUNT.
002240  020110  011337  003432    5$:     MOV     (R3),GDDAT      ;EXPECTED DATA
002244  020114  011237  003434            MOV     (R2),BDDAT      ;GET DATA FROM BUFFER
002250  020120  023737  003432  003434    CMP     GDDAT,BDDAT
002256  020126  001436                    BEQ     7$              ;BR IF DATA IS RIGHT.
002260  020130  010237  003424            MOV     R2,TMP0         ; GET ERROR ADDRESS...
002264  020134  010137  003426            MOV     R1,TMP1         ;...AND WORD NUMBER.
002270  020140  005737  003430            TST     TMP2            ; 1ST ERROR ENCOUNTERED ??
002274  020144  001004                    BNE     6$              ;NO, SKIP THE ERROR HEADER.
002276  020146                            DFERR   EM22,ERR0       ;ERROR MESSAGE ON 1ST ERROR...
    (5) 020146  104455                                                      TRAP    C$ERDF
    (6) 020150  000230                                                      .WORD   152
    (6) 020152  032037                                                      .WORD   EM22
    (6) 020154  026164                                                      .WORD   ERR0
002302  020156  005237  003430    6$:     INC     TMP2            ;...FOLLOWED BY FAILING DATA.
002306  020162                            PRINTX  #FRMT14,TMP1,TMP0,GDDAT,BDDAT
   (11) 020162  013746  003434                                              MOV     BDDAT,-(SP)
   (10) 020166  013746  003432                                              MOV     GDDAT,-(SP)
    (9) 020172  013746  003424                                              MOV     TMP0,-(SP)
    (8) 020176  013746  003426                                              MOV     TMP1,-(SP)
    (7) 020202  012746  030031                                              MOV     #FRMT14,-(SP)
```

```
                                                                                                    MOV     #5,-(SP)
                                                                                                    MOV     SP,R0
                                                                                                    TRAP    C$PNTX
                                                                                                    ADD     #14,SP
                        CKLOOP
                                                                                                    TRAP    C$CLP1
7$:                     TST     (R2)+           ;INCREMENT BUFFER POINTERS.
                        TST     (R3)+
                        INC     R1              ;AND THE WORD COUNT.
                        CMP     R1,#255.        ; DONE ALL WORDS ??
                        BLT     5$              ; NOT YET, CONTINUE.
                        TST     TMP2            ; YES, ANY ERRORS LOGGED ??
                        BEQ     8$              ;NO.
                        PRINTB  #FRMT15,TMP2    ;YES, PRINT SUMMARY.
                                                                                                    MOV     TMP2,-(SP)
                                                                                                    MOV     #FRMT15,-(SP)
                                                                                                    MOV     #2,-(SP)
                                                                                                    MOV     SP,R0
                                                                                                    TRAP    C$PNTB
                                                                                                    ADD     #6,SP
8$:                     ENDSEG
                                                                                            10000$:
                                                                                                    TRAP    C$ESEG
                        SETPRI  #PRI07
                                                                                                    MOV     #PRI07,R0
                                                                                                    TRAP    C$SPRI
                        RTS     PC              ; RETURN.

                        ENDTST
                                                                                            L10050:
                                                                                                    TRAP    C$ETST
```

```
  (6)  020206  012746  000005
  (3)  020212  010600
  (4)  020214  104415
  (4)  020216  062706  000014
2169   020222
  (3)  020222  104406
2170   020224  005722
2171   020226  005723
2172   020230  005201
2173   020232  020127  000377
2174   020236  002724
2175   020240  005737  003430
2176   020244  001412
2177   020246
  (8)  020246  013746  003430
  (7)  020252  012746  030120
  (6)  020256  012746  000002
  (3)  020262  010600
  (4)  020264  104414
  (4)  020266  062706  000006
2178   020272
  (3)  020272
  (3)  020272  104405
2179   020274
  (3)  020274  012700  000340
  (3)  020300  104441
2180   020302  000207
2181
2182   020304
  (3)  020304
  (3)  020304  104401
```

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

020306                          STARS
                                ;;*************************************************************************
                                ;* FOR RLV12:
                                ;* TEST THAT BBS7 WILL SELECT THE I/O PAGE AND THAT ACCESS TO
                                ;* LOCATION 0 IN THAT PAGE WILL GENERATE NXM AND OPI ERRORS.
                                ;* NOTE -- IF BANK 7 IS NOT PROPERLY SELECTED, 1000 BYTES
                                ;*         STARTING AT XXXX1000 WILL PROBABLY GET CPUNCHED !!!!
                                ;* FOR RLV11:
                                ;* RLV11 DOESN'T ASSERT BBS7, SO WE'LL THIS TEST
                                ;* IF PHYSICAL MEMORY SIZE IS 124K OR GREATER.
020306                          STARS
                                ;;*************************************************************************
020306                                  BEGIN.TEST
020306                                                                                  T28::
020306  023727  003402  000001        CMP     RLTYP,#RLV11
020314  003006                        BGT     1$              ; EXECUTE ALWAYS ON RLV12...
020316  023727  011232  000174        CMP     .MSIZE,#124.
020324  002406                        BLT     2$              ;...AND ON RLV11 IF < 124KW.
020326                                EXIT    TST
020326  104432                                                                  TRAP    C$EXIT
020330  000150                                                                  .WORD   L10053-.
020332  112737  000054  034112  1$:   MOVB    #',,EM80X       ; ADJUST ERROR TEXT FOR RLV12...
020340  000402                        SKP2
020342  105037  034112          2$:   CLRB    EM80X           ;...OR FOR RLV11.
020346  004537  010714                JSR     R5,CALCRC       ; DO CRC FOR INITIAL...
020352  000000                        .WORD   0               ;...DA OF 0.
020354  012737  160000  010352        MOV     #160000,BA16    ; POINT TO LOC 0 IN LAST 4K.
020362  012700  000003                MOV     #3,R0           ; SET AN 18 BIT...
020366  023727  011240  000026        CMP     .ABUSW,#22.
020374  001002                        BNE     3$
020376  012700  000077                MOV     #77,R0          ;...OR 22 BIT EXTENSION.
020402  010037  010354          3$:   MOV     R0,BA22
020406  004537  010210                JSR     R5,LDFUN        ; XCT MAINTENANCE...
020412  000000  177001                MAINT,  -511.           ;...WITH VALID WORD COUNT.
020416  004537  010640          BB7:  JSR     R5,WTCRDY
020422  012700  120000                MOV     #ERR!NXM,R0     ; EXPECT AT LEAST THESE ERRORS.
020426  043700  003364                BIC     E.CS,R0
020432  001415                        BEQ     4$              ; EXIT IF WE'RE OK.

020434  004537  007510                JSR     R5,GETERR       ; OTHERWISE, GET WHAT'S THERE.
020440  000240                        NOP
020442  013737  010354  003426        MOV     BA22,TMP1       ; AND GET PHYSICAL ADDRESS.
020450  013737  010352  003430        MOV     BA16,TMP2
020456                                DFERR   EM80,ERR12      ; BBS7 AND/OR NXM FAILURE.
020456  104455                                                                  TRAP    C$ERDF
020460  000231                                                                  .WORD   153
020462  034061                                                                  .WORD   EM80
020464  026500                                                                  .WORD   ERR12

020466  012737  004364  010352  4$:   MOV     #BUF1,BA16      ; RESET BA16 AND 22.
020474  005037  010354                CLR     BA22
020500                                ENDTST
020500                                                                  L10053:
020500  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL   29 -- RLV11/12 MAINTENANCE, EXTENDED MEMORY ACCESS TEST.

                                STARS
                                ;;*********************************************************************
                                ;* BYPASS THIS TEST IF NO KT SUPPORT.
                                ;* OTHERWISE, MEMORY HAS ALREADY BEEN SIZED.
                                ;*PAR'S 0 TO 6 ARE MAPPED TO THE 1ST 28K, AND PAR7 TO THE I/O PAGE.
                                ;*
                                ;*NOW ACCESS EXTENDED MEMORY IN 4K INCREMENTS.  STORE A RANDOM DATA
                                ;*PATTERN IN BUF1 (RELOCATED).  EXECUTE MAINTENANCE MODE, AND CHECK
                                ;*BUF2 (RELOCATED) FOR CORRECT DATA TRANSFER.  ALSO CHECK THAT THE FINAL
                                ;*BUS ADDRESS INCREMENTED CORRECTLY (OVERFLOWING INTO THE BAE).
                                ;*WHEN THE END OF PHYSICAL MEMORY IS REACHED, A BUS-ERROR WILL OCCUR
                                ;*DURING THE BUFFER SET-UP PHASE AND WE'LL LOOK FOR A NXM ERROR AT
                                ;*THE SAME PHYSICAL ADDRESS.  CONTINUE UNTIL WE REACH THE I/O PAGE
                                ;*AT 760000 (18 BIT), OR 17760000 (22 BIT).
                                STARS
                                ;;*********************************************************************
                                PAR4=   KIPAR0+10               ; USE KIPAR4 FOR ACCESSING.
                                        BEGIN.TEST
                                                                                        T29::
                                        CLR     TRPFLG          ; CLEAR TRAP FLAG.
                                        TST     MMR0            ; KT AVAILABLE ??
                                        240
                                        TST     TRPFLG
                                        BEQ     EXMTST          ; PROCEED IF SO...
                                1$:     EXIT    TST             ;...EXIT OTHERWISE.
                                                                                TRAP    C$EXIT
                                                                                .WORD   L10054-.
                                EXMTST: MOV     #7400,IOPAG     ; ASSUME 18 BITS, SET LIMIT AT 124K...
                                        CMP     .ABUSW,#22.     ; 22 BITS AVAILABLE ??
                                        BNE     1$              ; BR IF NOT.
                                        CMP     RLTYP,#RLV12X   ; CAN WE HANDLE THEM ALL ??
                                        BNE     1$              ; BR IF NOT.
                                        MOV     #20,MMR3        ; YES, SET KT IN 22 BIT MODE...
                                        MOV     #177400,IOPAG   ;...AND RAISE LIMIT TO 2044K.
                                1$:     CLR     BA22
                                        MOV     #177000,BA16    ; SET PHYSICAL ADDRESS AT 32K-256 WORDS.
                                        MOV     #1770,PAR4      ; SET PAR4 TO THE SAME POINT.
                                EXMLUP: JSR     R5,SETRAN       ; RANDOM DATA TO BUF1, ZERO BUF2.
                                        BGNSEG
                                                                                TRAP    C$BSEG
                                        MOV     #BUF1,R1        ; SETUP TO COPY FROM BUF1...
                                        MOV     #100000,R2      ;...TO (PAR4)+0
                                        MOV     #512.,R3
                                        MOV     #22$,ERRVEC     ; SOONER OR LATER WE'LL TRAP.
                                        CLR     TRPFLG
                                        INC     MMR0            ;****** KT ON ******************
                                2$:     MOV     (R1)+,(R2)+     ; COPY BUFFERS VIA PAR4.        *
                                        240,240                 ;                               *
                                        SOB     R3,2$           ; LOOP 'TIL DONE.               *
                                        BR      3$              ;                               *
                                22$:    MOV     R2,TRPFLG       ; NXM TRAP, SAVE ADDRESS...     *
                                        MOV     #3$,(SP)        ;                               *
                                        RTI                     ;...AND CONTINUE.               *
                                3$:     CLR     MMR0            ;****** KT OFF ******************
```

```
                                                                L 7
 2280  020700  012737  010624  000004          MOV     #TRAP4,ERRVEC   ; RESET TRAP.
 2281  020706  004537  010714                  JSR     R5,CALCRC       ; JUST TO BE CONSISTENT.
 2282  020712  000000                          0
 2283  020714  004537  010210                  JSR     R5,LDFUN
 2284  020720  000000  177001                  MAINT, -511.            ; EXECUTE MAINT MODE DMA TO EXT MEM.
 2285  020724  004537  010640                  JSR     R5,WTCRDY
 2286  020730  023727  003402  000003          CMP     RLTYP,#RLV12X   ; BAE IN USE ??
 2287  020736  001411                          BEQ     33$             ; BR IF SO.
 2288  020740  013700  003364                  MOV     E.CS,R0         ; NO, GET EA<17:16> FROM FINAL CSR.
 2289  020744  006200                          ASR     R0
 2290  020746  006200                           ASR    R0
 2291  020750  006200                            ASR   R0
 2292  020752  006200                             ASR  R0
 2293  020754  042700  177774                  BIC     #^C3,R0
 2294  020760  000402                          SKP2
 2295  020762  013700  003376          33$:    MOV     E.BAE,R0        ; GET EA<21:16> FROM BAE.
 2296  020766  010037  003426                  MOV     R0,TMP1         ; SAVE FINAL PA IN CASE...
 2297  020772  013737  003366  003430          MOV     E.BA,TMP2       ;...WE HAVE A STATUS ERROR.
 2298  021000  005737  010626                  TST     TRPFLG          ; NOW, ARE WE EXPECTING NXM ??
 2299  021004  001415                          BEQ     4$              ; BR IF NOT.
 2300  021006  012700  120000                  MOV     #ERR!NXM,R0     ; YES, DO WE HAVE IT ??
 2301  021012  043700  003364                  BIC     E.CS,R0
 2302  021016  001501                          BEQ     8$              ; BR IF SO.
 2303  021020  004537  007510                  JSR     R5,GETERR       ; NO, GET WHATEVER'S THERE.
 2304  021024  000240                          NOP
 2305  021026                                  DFERR   EM90,ERR12      ; EXTENDED NXM STATUS WRONG.
   (5) 021026  104455                                                                    TRAP    C$ERDF
   (6) 021030  000232                                                                    .WORD   154
   (6) 021032  034147                                                                    .WORD   EM90
   (6) 021034  026500                                                                    .WORD   ERR12
 2306  021036  000471                          BR      8$
 2307
 2308  021040  004537  007510          4$:     JSR     R5,GETERR       ; ANY OTHER STATUS ERRORS ??
 2309  021044  000405                          BR      44$             ; BR IF NOT.
 2310  021046                                  DFERR   EM91,ERR13      ; STATUS INCORRECT ON EXT MEMORY ACCESS.
   (5) 021046  104455                                                                    TRAP    C$ERDF
   (6) 021050  000233                                                                    .WORD   155
   (6) 021052  034223                                                                    .WORD   EM91
   (6) 021054  026534                                                                    .WORD   ERR13
 2311  021056  000461                          BR      8$
 2312
 2313  021060  012701  100000          44$:    MOV     #100000,R1      ; POINT TO BUF1 IN EXT MEMORY.
 2314  021064  012702  101000                  MOV     #101000,R2      ; DITTO BUF2.
 2315  021070  005003                          CLR     R3              ; WORD NUMBER.
 2316  021072  012704  000377                  MOV     #255.,R4        ; LOOP COUNT.
 2317  021076  005237  177572                  INC     MMR0            ;****** KT ON ********************
 2318  021102  012137  003432          5$:     MOV     (R1)+,GDDAT     ;                              *
 2319  021106  012237  003434                  MOV     (R2)+,BDDAT     ;                              *
 2320  021112  023737  003432  003434          CMP     GDDAT,BDDAT     ; COMPARE DATA.                *
 2321  021120  001002                          BNE     6$              ; BR IF WRONG.                 *
 2322  021122  005203                          INC     R3              ; BUMP WORD COUNT.             *
 2323  021124  077412                          SOB     R4,5$           ; LOOP.                        *
 2324  021126  005037  177572          6$:     CLR     MMR0            ;****** KT OFF *******************
 2325  021132  005704                          TST     R4              ; LOOP COMPLETED (NO ERRORS) ??
 2326  021134  001432                          BEQ     8$              ; BR IF SO.
 2327  021136  010337  003424                  MOV     R3,TMP0         ; NO, SAVE FAILING WORD NUMBER.
```

```
 2328  021142  012703  000006            MOV     #6,R3
 2329  021146  013737  172350  003430    MOV     PAR4,TMP2
 2330  021154  005037  003426            CLR     TMP1            ; NOW ASSEMBLE BAD PA,
 2331  021160  006337  003430    7$:     ASL     TMP2            ; ADJUST PAGE TO BIT 21...
 2332  021164  006137  003426             ROL     TMP1
 2333  021170  077305                       SOB     R3,7$
 2334  021172  042702  160000            BIC     #^C17777,R2     ;...STRIP PAR BITS FROM VA...
 2335  021176  162702  000002            SUB     #2,R2
 2336  021202  060237  003430            ADD     R2,TMP2         ;...AND COMBINE THE PIECES.
 2337  021206  005537  003426             ADC     TMP1
 2338  021212                            DFERR   EM92,ERR14      ; DATA ERROR IN EXTENDED MEM.
   (5) 021212  104455                                                            TRAP    C$ERDF
   (6) 021214  000234                                                            .WORD   156
   (6) 021216  034300                                                            .WORD   EM92
   (6) 021220  026544                                                            .WORD   ERR14
 2339
 2340  021222                      8$:     CKLOOP                                          TRAP    C$CLP1
   (3) 021222  104406
 2341  021224                            ENDSEG                                  10000$:
   (3) 021224                                                                    TRAP    C$ESEG
   (3) 021224  104405
 2342  021226  062737  020000  010352    ADD     #20000,BA16     ; INCR PHYSICAL ADDRESS BY 4K...
 2343  021234  005537  010354             ADC     BA22            ;...AND OVERFLOW INTO BAE.
 2344  021240  062737  000200  172350    ADD     #200,PAR4       ; INCR PAR4 BY 4K ALSO.
 2345  021246  023727  172350    9$:     CMP     PAR4,(PC)+      ; QUIT AT THE I/O PAGE...
 2346  021252  007400             IOPAG:  7400                    ;...007400 (124K)...
 2347                                                            ;...OR 177400 (2044K).
 2348  021254  103002                     BHIS    EXMDUN          ; EXIT IF DONE.
 2349  021256  000137  020610             JMP     EXMLUP          ; LOOP OTHERWISE.
 2350
 2351  021262  012737  004364  010352  EXMDUN:  MOV     #BUF1,BA16      ; RESET BA16 AND 22.
 2352  021270  005037  010354             CLR     BA22
 2353  021274                            ENDTST                                  L10054:
   (3) 021274                                                                    TRAP    C$ETST
   (3) 021274  104401
```

```
                                        .SBTTL
                                        .SBTTL  RLV12 DRIVE INTERFACE TESTS (G5388 TLM REQUIRED).
                                        .SBTTL
021276                                  $TARS
                                        ;;*************************************************************************
                                        ;* ALL THESE TESTS REQUIRE THE SERVICES OF THE G5388 TEST LOOP MODULE.
                                        ;*
                                        .SBTTL   30 -- SYS CLK, PWR OK, DRIVE SELECT, READY, AND ERROR BITS.
                                        ;*
                                        ;* 1.  TEST SYS CLOCK AND PWR OK FROM THE RLV12.
                                        ;* 2.  TEST DRIVE READY AND DRIVE ERROR TO THE RLV12.
                                        ;* 3.  TEST DRIVE SELECT BITS FROM THE RLV12.
021276                                  $TARS
                                        ;;*************************************************************************
021276                                  TLM1:   BEGIN.TEST
021276                                                                                          T30::
021276  004737  023762                          JSR     PC,TLMOK        ; CHECK FOR V12 AND TLM OK...
021302  102561                                BVS     7$              ;...AND EXIT IF NOT.
021304  012777  000001  162032                MOV     #1,@TCSR        ; RESET...
021312  005077  162026                        CLR     @TCSR           ;...AND CLEAR TLM CSR (LO BYTE).
021316  017737  162022  003434                MOV     @TCSR,BDDAT     ; GET TLM STATUS.
021324  012737  004400  003432                MOV     #4400,GDDAT     ; EXPECT SYSCLK<11> AND PWROK<8>.
021332  023737  003432  003434                CMP     GDDAT,BDDAT
021340  001404                                BEQ     1$              ; BR IF STATUS IS RIGHT.
021342                                        DFERR   EM101,ERR2      ; PWR-OK AND/OR SYS-CLK NOT SET IN TLM.
021342  104455                                                                    TRAP    C$ERDF
021344  000235                                                                    .WORD   157
021346  035007                                                                    .WORD   EM101
021350  026220                                                                    .WORD   ERR2
021352  017700  161746          1$:     MOV     @RLCS,R0        ; GET RLV12 CSR.
021356  010037  003434                  MOV     R0,BDDAT        ; REC'D RLV12 STATUS...
021362  042700  000001                  BIC     #DRDY,R0        ;...SHOULD HAVE DRDY CLEAR.
021366  010037  003432                  MOV     R0,GDDAT
021372  023737  003432  003434          CMP     GDDAT,BDDAT
021400  001404                          BEQ     2$              ; BR IF SO.
021402                                  DFERR   EM102,ERR2      ; DRIVE READY NOT CLEAR IN RLV12.
021402  104455                                                                    TRAP    C$ERDF
021404  000236                                                                    .WORD   158
021406  035070                                                                    .WORD   EM102
021410  026220                                                                    .WORD   ERR2
021412  012777  000200  161724  2$:     MOV     #200,@TCSR      ; SET DRIVE READY IN TLM.
021420  017700  161700                  MOV     @RLCS,R0        ; GET RLV12 AGAIN.
021424  010037  003434                  MOV     R0,BDDAT
021430  052700  000001                  BIS     #DRDY,R0        ; DRDY SHOULD BE SET NOW.
021434  010037  003432                  MOV     R0,GDDAT
021440  023737  003432  003434          CMP     GDDAT,BDDAT
021446  001404                          BEQ     3$              ; BR IF SO.
021450                                  DFERR   EM102,ERR2      ; DRIVE READY NOT SET IN RLV12.
021450  104455                                                                    TRAP    C$ERDF
021452  000237                                                                    .WORD   159
021454  035070                                                                    .WORD   EM102
021456  026220                                                                    .WORD   ERR2
021460  012777  000100  161656  3$:     MOV     #100,@TCSR      ; CLEAR DRDY, SET DERR IN TLM.
021466  017700  161632                  MOV     @RLCS,R0        ; ONE MORE TIME.
021472  010037  003434                  MOV     R0,BDDAT
021476  052700  140000                  BIS     #ERR!DERR,R0    ; SHOULD HAVE THESE ERROR BITS.
```

```
2396  021502  010037  003432                    MOV     R0,GDDAT
2397  021506  023737  003432  003434            CMP     GDDAT,BDDAT
2398  021514  001404                            BEQ     4$                      ; BR IF SO.
2399  021516                                    DFERR   EM103,ERR2              ; DRIVE ERROR STATUS INCORRECT IN RLV12.
  (5) 021516  104455                                                                    TRAP    C$ERDF
  (6) 021520  000240                                                                    .WORD   160
  (6) 021522  035145                                                                    .WORD   EM103
  (6) 021524  026220                                                                    .WORD   ERR2
2400  021526  005077  161612          4$:       CLR     @TCSR                   ; CLEAR TLM.
2401  021532  005001                            CLR     R1                      ; INIT DRIVE SELECT BITS<9:8>...
2402  021534  112737  000060  035235            MOVB    #'0,EM104X              ;...AND ERROR TEXT.
2403  021542  012703  000004                    MOV     #4,R3                   ; LOOP CONTROL.
2404  021546  012777  000200  161550  5$:       MOV     #CRDY,@RLCS             ; INIT RLV CSR...
2405  021554  050177  161544                    BIS     R1,@RLCS                ;...INSERT DRIVE BITS <9:8>.
2406  021560  017702  161560                    MOV     @TCSR,R2                ; GET TLM CSR.
2407  021564  010237  003434                    MOV     R2,BDDAT                ; RECV'D TLM SHOULD ECHO DRIVE...
2408  021570  042702  003000                    BIC     #BIT10!BIT9,R2          ;...SELECTED IN TLM<10:9>.
2409  021574  006301                            ASL     R1
2410  021576  050102                             BIS    R1,R2                   ; SET EXPECTED VALUE.
2411  021600  006201                              ASR   R1
2412  021602  010237  003432                    MOV     R2,GDDAT
2413  021606  023737  003432  003434            CMP     GDDAT,BDDAT
2414  021614  001404                            BEQ     6$
2415  021616                                    DFERR   EM104,ERR2              ; DRIVE SELECT FAILS IN TLM.
  (5) 021616  104455                                                                    TRAP    C$ERDF
  (6) 021620  000241                                                                    .WORD   161
  (6) 021622  035227                                                                    .WORD   EM104
  (6) 021624  026220                                                                    .WORD   ERR2
2416  021626  062701  000400          6$:       ADD     #DS1,R1                 ; BUMP DRIVE SELECT NUMBER...
2417  021632  105237  035235                    INCB    EM104X                  ;...AND THE ERROR TEXT.
2418  021636  077335                            SOB     R3,5$
2419  021640  012777  000200  161456            MOV     #CRDY,@RLCS             ; RESET DRIVE NUMBER IN RL.
2420  021646                          7$:       ENDTST
  (3) 021646                                                                    L10055:
  (3) 021646  104401                                                                    TRAP    C$ETST
```

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

                                STARS
                                ;;*************************************************************************
                                ;* NOW TEST DRIVE COMMAND, STATUS, AND STATUS CLOCK
                                ;* USING A "GET STATUS" COMMAND.
                                STARS
                                ;;*************************************************************************
                                TLM2:   BEGIN.TEST
                                                                                        T31::
                                        JSR     PC,TLMOK
                                        BVS     5$                      ; EXIT IF NO GOOD.
                                        JSR     PC,CLRTLM               ; RESET TLM, AND SET DRDY.
                                        CLR     INIMP
                                        MOV     #125252,4(R5)   ; SET PHONEY STATUS WORD.
                                        MOV     #CRDY!GSTAT,@RLCS ; SET "GET STATUS" COMMAND...
                                        MOV     #DRST!GSBIT!MK,@RLDA ;...AND RESET STATUS BITS.
                                        JSR     R5,BEFORE
                                        BIC     #CRDY,@RLCS     ; XCT IT.
                                        CLR     R0
                                1$:     MOV     (R5),R1                 ; GET TCSR...
                                        BMI     2$                      ;...PROCEED WHEN "NEW SKGS" SETS...
                                        SOB     R0,1$                   ;...BUT DON'T WAIT FOREVER !!!
                                2$:     MOV     2(R5),BDDAT     ; SAVE RECV'D DRIVE COMMAND (SKGS)...
                                        JSR     R5,WTCRDY
                                        MOV     #DRST!GSBIT!MK,GDDAT ;...SHOULD LOOK LIKE THIS.
                                        CMP     GDDAT,BDDAT
                                        BEQ     3$                      ; BR IF STATUS COMMAND WAS RIGHT.
                                        DFERR   EM105,ERR4      ; GET STATUS COMMAND WRONG IN TLM.
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   162
                                                                                        .WORD   EM105
                                                                                        .WORD   ERR4
                                3$:     MOV     E.MP,BDDAT      ; RETURNED DRIVE STATUS...
                                        MOV     #125252,GDDAT   ;...SHOULD = PHONEY.
                                        CMP     GDDAT,BDDAT
                                        BEQ     4$              ; BR IF RETURNED STATUS IS RIGHT.
                                        DFERR   EM106,ERR4      ;RETURNED STATUS INCORRECT IN RLMP.
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   163
                                                                                        .WORD   EM106
                                                                                        .WORD   ERR4
                                4$:     JSR     R5,GETERR       ; FINAL CONTROLLER STATUS...
                                        BR      5$              ;...SHOULD BE ERROR FREE.
                                        MOV     #EXPNON,TMP1
                                        DFERR   EM107,ERR20A    ; RLV12 ERRORS AFTER RESET GET STATUS
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   164
                                                                                        .WORD   EM107
                                                                                        .WORD   ERR20A
                                5$:     ENDTST
                                                                                L10056:
                                                                                        TRAP    C$ETST
```

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

                                        STARS
                                        ;;*************************************************************************
                                        ;* TEST SECTOR PULSE USING A "SEEK" COMMAND.
                                        STARS
                                        ;;*************************************************************************
                                        TLM3:   BEGIN.TEST
                                                                                                        T32::
                                                JSR     PC,TLMOK
                                                BVS     4$              ; BR IF NOT OK.
                                                JSR     PC,CLRTLM       ; RESET TTLM, SET DRDY.
                                                CLR     INIMP
                                                BIS     #24,(R5)        ; ENABLE SECTOR GENERATOR (NOM CLOCK).
                                                MOV     #CRDY!SEEK,@RLCS ; SET "SEEK" COMMAND...
                                                MOV     #77600!MK,@RLDA ;...AND A PHONEY SEEK DIFFERENCE.
                                                JSR     R5,BEFORE
                                                BIC     #CRDY,@RLCS     ; XCT IT.
                                                CLR     R0
                                        1$:     MOV     (R5),R1
                                                BMI     2$              ; PROCEED WHEN "NEW SKGS" SETS...
                                                SOB     R0,1$           ;...KEEP-ALIVE !!
                                        2$:     MOV     2(R5),BDDAT     ; SAVE RECV'D DRIVE COMMAND...
                                                JSR     R5,WTCRDY
                                                MOV     #77600!MK,GDDAT ;...SHOULD = PHONEY DIFFERENCE.
                                                CMP     GDDAT,BDDAT
                                                BEQ     3$              ; BR IF SO.
                                                DFERR   EM110,ERR4      ; DRIVE COMMAND WRONG IN TLM.
                                                                                        TRAP    CSERDF
                                                                                        .WORD   165
                                                                                        .WORD   EM110
                                                                                        .WORD   ERR4
                                        3$:     JSR     R5,GETERR       ; FINAL CONTROLLER STATUS...
                                                BR      4$              ;...SHOULD BE ERROR FREE.
                                                MOV     #EXPNON,TMP1
                                                DFERR   EM111,ERR20A    ; RLV ERRORS AFTER SEEK COMMAND
                                                                                        TRAP    CSERDF
                                                                                        .WORD   166
                                                                                        .WORD   EM111
                                                                                        .WORD   ERR20A
                                        4$:     ENDTST
                                                                                L10057:
                                                                                        TRAP    CSETST
```

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

                                STARS
                                ;;*************************************************************************
                                ;* TEST THE WRITE GATE, WRITE DATA ACTIVE, AND WRITE GATE ERROR.
                                STARS
                                ;;*************************************************************************
                                TLM4:   BEGIN.TEST
                                                                                                T33::
                                        JSR     PC,TLMOK
                                        BVS     5$              ; BR IF NOT OK.
                                        JSR     PC,CLRTLM       ; RESET TLM, SET DRDY.
                                        BIS     #24,(R5)        ; ENABLE SECTOR GEN (NOM CLOCK).
                                        JSR     R5,WDELAY       ; DELAY SO THAT WE ENTER THE TLM...
                                        2                       ;...SEQUENCE ASYNCHRONOUSLY.
                                        MOV     #CRDY!WRITE,@RLCS ; SET COMMAND TO WRITE...
                                        MOV     #-128.,INIMP
                                        MOV     INIMP,@RLMP     ;...128 WORDS...
                                        MOV     #BUF1,@RLBA     ;...FROM BUF1...
                                        MOV     #205,@RLDA      ;...TO CYL 1, SECT 5.
                                        JSR     R5,BEFORE
                                        BIC     #CRDY,@RLCS     ; XCT IT.
                                        CLR     R0
                                1$:     MOV     (R5),R1         ; TLM STATUS => R1.
                                        MOV     R1,BDDAT        ; RECV'D TLM STATUS...
                                        BIS     #BIT13!BIT12,R1 ;...WILL GET WRITE GATE AND DATA ACTIVE.
                                        MOV     R1,GDDAT
                                        CMP     GDDAT,BDDAT
                                        BEQ     2$              ; BR IF AND WHEN THAT OCCURS...
                                        SOB     R0,1$           ;...WITHIN REASON -- OF COURSE.
                                        JSR     R5,WTCRDY
                                        DFERR   EM112,ERR4      ; WRITE GATE AND/OR DATA ACTIVE NOT SET.
                                                                                                TRAP    C$ERDF
                                                                                                .WORD   167
                                                                                                .WORD   EM112
                                                                                                .WORD   ERR4
                                        SKP2
                                2$:     JSR     R5,WTCRDY
                                        MOV     (R5),BDDAT      ; FINAL TLM STATUS...
                                        MOV     BDDAT,GDDAT
                                        BIC     #BIT14!BIT13,GDDAT ;...ERROR AND GATE SHOULD BE CLEAR.
                                        CMP     GDDAT,BDDAT
                                        BEQ     4$              ; BR IF SO.
                                        BIT     #BIT13,BDDAT    ; IT'S NOT, IS GATE STILL SET ??
                                        BEQ     3$              ;
                                        DFERR   EM113A,ERR4     ; WRITE GATE STILL SET AFTER DONE.
                                                                                                TRAP    C$ERDF
                                                                                                .WORD   168
                                                                                                .WORD   EM113A
                                                                                                .WORD   ERR4
                                        BR      4$
                                3$:     DFERR   EM113,ERR4      ; WRITE GATE ERROR IN TLM.
                                                                                                TRAP    C$ERDF
                                                                                                .WORD   169
                                                                                                .WORD   EM113
                                                                                                .WORD   ERR4
```

```
 2531  022454  004537  007510          4$:     JSR     R5,GETERR       ; FINAL RLV STATUS...
 2532  022460  000407                          BR      5$              ;...SHOULD BE ERROR FREE.
 2533  022462  012737  036670  003426          MOV     #EXPNON,TMP1
 2534  022470                                  DFERR   EM114,ERR20A    ; RL ERROR AFTER WRITE.
   (5) 022470  104455                                                                  TRAP    C$ERDF
   (6) 022472  000252                                                                  .WORD   170
   (6) 022474  036066                                                                  .WORD   EM114
   (6) 022476  026666                                                                  .WORD   ERR20A
 2535  022500                          5$:     ENDTST
   (3) 022500                                                                  L10060:
   (3) 022500  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL   34 -- READ DATA, READ HEADER, AND READ DATA W/O HEADER.

                                STARS
                                ;;*************************************************************************
                                ;* READ AND VERIFY DATA ON EACH OF 6 TLM SECTORS.
                                ;* SECTOR 3 GETS A READ HEADER AND READ DATA W/O HEADER SEQUENCE.
                                ;* REPEAT USING ALTERNATE VCO SETTINGS (WITH MAX AND MIN PEAK SHIFT).
                                ;* VARIOUS PEAK SHIFT, CLOCK, AND SECTOR OPTIONS MAY BE SELECTED VIA
                                ;* THE "CHANGE SOFTWARE" DIALOGUE AT START/RESTART TIME.
                                STARS
                                ;;*************************************************************************
                                TLM5:   BEGIN.TEST
                                                                                                T34::
                                        JSR     PC,TLMPID       ; VERIFY TLM OK AND PROM ID.
                                        BVS     8$              ; EXIT IF NOT OK.
                                        MOV     #-128.,INIMP    ; INIT WORD COUNT.
                                        MOV     SNGLSEC,12$     ; GET SECTOR NUMBER (IF ANY).
                                        BIC     #^C7,12$
                                        BNE     11$             ; NZ = SINGLE SECTOR MODE.
                                10$:    INC     12$             ; OTHERWISE, USE ALL STARTING WITH 1.
                                11$:    JSR     PC,PSX          ; SET BUFFERS AND DISK ADDRESS...
                                12$:    1                       ;...FOR THIS SECTOR NUMBER.
                                        BVC     1$              ; BR IF SECTOR FOUND.
                                        MOVB    TSECX,EM119X    ; SET ASCII SECTOR NUMBER.
                                        DFERR   EM119,NOSIG     ; CAN'T FIND SECTOR.
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   171
                                                                                        .WORD   EM119
                                                                                        .WORD   NOSIG
                                        BR      7$
                                1$:     BIT     #BIT2,MPXCLK    ; NOMINAL CLOCK ENABLED ??
                                        BNE     2$              ; BR IF NOT.
                                        MOV     #224,(R5)       ; YES, SET NOMINAL CLOCK, NO PEAK SHIFT.
                                        MOV     #TCLK0,TMP2
                                        BR      4$
                                2$:     BIT     #BIT1,MPXCLK    ; FAST CLOCK ENABLED ??
                                        BNE     3$              ; BR IF NOT.
                                        MOV     #230,(R5)       ; YES, SET FAST CLOCK, NO PEAK SHIFT.
                                        MOV     #TCLK1,TMP2
                                        BR      4$
                                3$:     BIT     #BIT0,MPXCLK    ; SLOW CLOCK ENABLED ??
                                        BNE     7$              ; BR IF NOT.
                                        MOV     #220,(R5)       ; YES, SET SLOW CLOCK, NO PEAK SHIFT.
                                        MOV     #TCLK2,TMP2

                                4$:     BIT     #BIT14,MPXCLK   ; MIN PEAK SHIFT ENABLED ??
                                        BNE     5$              ; BR IF NOT.
                                        CLRB    TPEAK           ; YES, PEAK SHIFT IS OFF...
                                        JSR     PC,READSEC      ;...READ SECTOR AND VERIFY DATA.
                                5$:     TST     MPXCLK          ; MAX PEAK SHIFT ENABLED ??
                                        BMI     6$              ; BR IF NOT.
                                        BIS     #40,(R5)        ; YES, MAX PEAK SHIFT ON...
                                        MOVB    #40,TPEAK       ;...EXTEND CLOCK MESSAGE...
                                        JSR     PC,READSEC      ;...DO IT ALL AGAIN.

                                6$:     BIT     #4,(R5)         ; NOMINAL DONE ??
```

```
 2586  022732  001330                        BNE     2$                ; LOOP TO "FAST" IF SO.
 2587  022734  032715  000010                BIT     #10,(R5)          ; FAST DONE ??
 2588  022740  001337                        BNE     3$                ; LOOP TO "SLOW" IF SO...
 2589                                                                   ;...OTHERWISE, THIS SECTOR IS DONE.
 2590
 2591  022742  032737  000007  003314  7$:   BIT     #7,SNGLSEC        ; SINGLE SECTOR MODE ??
 2592  022750  001004                        BNE     8$                ; WE'RE DONE IF SO.
 2593  022752  023727  022544  000006        CMP     12$,#6            ; NO, LAST SECTOR DONE ??
 2594  022760  103665                        BLO     10$               ; GO 'ROUND AGAIN IF NOT...
 2595  022762                          8$:   EXIT    TST               ;...OTHERWISE, NEXT TEST.
   (3) 022762  104432                                                                TRAP    C$EXIT
   (3) 022764  000376                                                                .WORD   L10061-.
 2596
 2597                                  ;
 2598                                  ; SUBROUTINE TO READ DATA FROM TLM TO BUF2 AND VERIFY (AGAINST BUF1).
 2599                                  ; DISK ADDRESS AND EXPECTED DATA (BUF1) MUST BE PRESET BEFORE CALL.
 2600                                  ; WORD COUNT (-128.) IS PRESET FOR ALL.
 2601  022766                          ;
                                      READSEC: BGNSEG
   (3) 022766  104404                                                                TRAP    C$BSEG
 2602  022770  012700  005364                MOV     #BUF2,R0
 2603  022774  012701  000200                MOV     #128.,R1
 2604  023000  012720  177777          1$:   MOV     #-1,(R0)+         ; CLEAR RECEIVING BUFFER.
 2605  023004  077103                        SOB     R1,1$
 2606  023006  105037  036205                CLRB    EM115X            ; INIT ERROR TEXT FOR "READ".
 2607  023012  105037  036355                CLRB    EM117X
 2608  023016  012777  000214  160300        MOV     #CRDY!READ,@RLCS  ; SET COMMAND TO READ...
 2609  023024  013777  003436  160300        MOV     INIMP,@RLMP       ;...128 WORDS...
 2610  023032  013777  003440  160270        MOV     INIDA,@RLDA       ;...FROM CYL X, SECT Y...
 2611  023040  012777  005364  160260        MOV     #BUF2,@RLBA       ;...TO BUF2.
 2612  023046  004537  010016          2$:   JSR     R5,BEFORE
 2613  023052  042777  000200  160244        BIC     #CRDY,@RLCS       ; XCT.
 2614  023060  004537  010640                JSR     R5,WTCRDY         ; WAIT FOR RL DONE.
 2615  023064  123737  024506  003365        CMPB    PSERR,E.CS+1      ; FINAL STATUS RIGHT ??
 2616  023072  001412                        BEQ     3$                ; BR IF SO.
 2617  023074  004537  007510                JSR     R5,GETERR         ; NO, GET WHATEVER'S THERE.
 2618  023100  000240                        NOP
 2619  023102  013737  024504  003426        MOV     PSETXT,TMP1
 2620  023110                                DFERR   EM115,ERR20       ; ERROR STATUS WRONG ON READ.
   (5) 023110  104455                                                                TRAP    C$ERDF
   (6) 023112  000254                                                                .WORD   172
   (6) 023114  036137                                                                .WORD   EM115
   (6) 023116  026662                                                                .WORD   ERR20
 2621  023120  023727  024504  036624  3$:   CMP     PSETXT,#EXPHCRC   ; NOW IS THIS THE BAD HEADER SECTOR ??
 2622  023126  001062                        BNE     CHKDATA           ; BR IF NOT.
 2623
 2624  023130  012700  000007          READNH: MOV     #7,R0             ; YES IT IS, SET UP TO READ HEADERS.
 2625  023134  112737  000040  036205        MOVB    #40,EM115X        ; EXTEND ERROR TEXT TO "READ NH"
 2626  023142  112737  000040  036355        MOVB    #40,EM117X
 2627  023150  012777  000210  160146  1$:   MOV     #CRDY!RDHDR,@RLCS
 2628  023156  042777  000200  160140        BIC     #CRDY,@RLCS       ; XCT READ HEADER...
 2629  023164  105777  160134          2$:   TSTB    @RLCS             ;...AND WAIT FOR DONE.
 2630  023170  100375                        BPL     2$
 2631  023172  027727  160134  000206        CMP     @RLMP,#<1_7.>!6   ; FOUND BAD SECTOR-1 ??
 2632  023200  001406                        BEQ     3$                ; BR IF SO.
 2633  023202  077016                        SOB     R0,1$
 2634  023204                                DFERR   EM116,ERR21       ; CAN'T FIND CYL 1, SECT 6
```

```
                                                                                    TRAP    C$ERDF
                                                                                    .WORD   173
                                                                                    .WORD   EM116
                                                                                    .WORD   ERR21
                        BR      READDUN         ; ABORT THE REST.
3$:                     MOV     #CRDY!RDNHDR,@RLCS ; OK, SET READ-NO-HEADER COMMAND...
                        MOV     INIMP,@RLMP     ;...INSURE WORD COUNT IS STILL THERE...
                        JSR     R5,BEFORE
                        BIC     #CRDY,@RLCS     ;...AND XCT IT.
                        JSR     R5,WTCRDY
                        JSR     R5,GETERR       ; SHOULD HAVE NO ERRORS...
                        BR      CHKDATA         ;...BR IF SO.
                        MOV     #EXPNON,TMP1
                        DFERR   EM115,ERR20     ; ERROR STATUS INCORRECT AFTER READ NH
                                                                                    TRAP    C$ERDF
                                                                                    .WORD   174
                                                                                    .WORD   EM115
                                                                                    .WORD   ERR20
CHKDATA:                MOV     #BUF1+48.,R0    ; EXPD DATA FIELD BEGINS AT WORD 24...
                        MOV     #BUF2,R1        ;...RECVD DATA AT WORD 0.
                        CLR     R2              ;...AND WORD NUMBER.
1$:                     CMP     (R0),(R1)       ; COMPARE EXP'D VS REC'D DATA...
                        BEQ     2$              ;...AND BR IF OK.
                        MOV     (R0),GDDAT      ; IT'S NOT, SET EXP'D...
                        MOV     (R1),BDDAT      ;...RECV'D...
                        MOV     R1,TMP0         ;...ADDRESS...
                        MOV     R2,TMP1         ;...AND WORD NUMBER.
                        DFERR   EM117,ERR22     ; DATA INCORRECT ON READ OR READ NH.
                                                                                    TRAP    C$ERDF
                                                                                    .WORD   175
                                                                                    .WORD   EM117
                                                                                    .WORD   ERR22
                        BR      3$
2$:                     CMP     (R0)+,(R1)+     ; BUMP ADDRESSES...
                        INC     R2              ;...AND WORD NUMBER.
                        BIT     #177,R2
                        BNE     1$              ;...AND LOOP 'TIL DONE.
3$:
READDUN:                ENDSEG
                                                                            10000$:
                                                                                    TRAP    C$ESEG
                        RTS     PC
                        ENDTST
                                                                            L10061:
                                                                                    TRAP    C$ETST
```

```
                                .SBTTL   35 -- WRITE CHECK.

                                STARS
                                ;;*******************************************************************
                                ;* DO A WRITE CHECK USING SECTOR AND CLOCK OPTIONS AS BEFORE.
                                STARS
                                ;;*******************************************************************
                                TLM6:   BEGIN.TEST
                                                                                        T35::
                                        JSR     PC,TLMPID
                                        BVS     8$              ; EXIT IF NOT OK.
                                        MOV     #-128.,INIMP    ; INIT WORD COUNT.
                                        MOV     SNGLSEC,12$     ; GET SECTOR NUMBER (IF ANY).
                                        BIC     #^C7,12$
                                        BNE     11$             ; NZ = SINGLE SECTOR MODE.
                                10$:    INC     12$             ; OTHERWISE, USE ALL STARTING WITH 1.
                                11$:    JSR     PC,PSX          ; SET BUFFERS AND DISK ADDRESS...
                                12$:    1                       ;...FOR THIS SECTOR NUMBER.
                                        BVC     1$
                                        MOVB    TSECX,EM119X    ; SET ASCII SECTOR NUMBER.
                                        DFERR   EM119,NOSIG     ; CAN'T FIND SECTOR.
                                                                                        TRAP    C$ERDF
                                                                                        .WORD   176
                                                                                        .WORD   EM119
                                                                                        .WORD   NOSIG
                                        BR      7$
                                1$:     BIT     #BIT2,MPXCLK    ; NOMINAL CLOCK ENABLED ??
                                        BNE     2$              ; BR IF NOT.
                                        MOV     #224,(R5)       ; YES, SET NOMINAL CLOCK, NO PEAK SHIFT.
                                        MOV     #TCLK0,TMP2
                                        BR      4$
                                2$:     BIT     #BIT1,MPXCLK    ; FAST CLOCK ENABLED ??
                                        BNE     3$              ; BR IF NOT.
                                        MOV     #230,(R5)       ; YES, SET FAST CLOCK, NO PEAK SHIFT.
                                        MOV     #TCLK1,TMP2
                                        BR      4$
                                3$:     BIT     #BIT0,MPXCLK    ; SLOW CLOCK ENABLED ??
                                        BNE     7$              ; BR IF NOT.
                                        MOV     #220,(R5)       ; YES, SET SLOW CLOCK, NO PEAK SHIFT.
                                        MOV     #TCLK2,TMP2

                                4$:     BIT     #BIT14,MPXCLK   ; MIN PEAK ENABLED ??
                                        BNE     5$              ; BR IF NOT.
                                        CLRB    TPEAK           ; YES, PEAK SHIFT IS OFF...
                                        JSR     PC,WRITCHK      ;...WRITE-CHECK THIS SECTOR.
                                5$:     TST     MPXCLK          ; MAX PEAK SHIFT ENABLED ??
                                        BMI     6$              ; BR IF NOT.
                                        BIS     #40,(R5)        ; YES, MAX PEAK SHIFT ON...
                                        MOVB    #40,TPEAK       ;...EXTEND CLOCK MESSAGE...
                                        JSR     PC,WRITCHK      ;...AND DO IT AGAIN.

                                6$:     BIT     #4,(R5)         ; NOMINAL DONE ??
                                        BNE     2$              ; LOOP TO "FAST" IF SO.
                                        BIT     #10,(R5)        ; FAST DONE ??
                                        BNE     3$              ; LOOP TO "SLOW" IF SO...
                                                                ;...OTHERWISE, THIS SECTOR IS DONE.
```

```
 2715
 2716  023624  032737  000007  003314  7$:     BIT     #7,SNGLSEC      ; SINGLE SECTOR MODE ??
 2717  023632  001004                          BNE     8$              ; WE'RE ALL DONE IF SO.
 2718  023634  023727  023426  000006          CMP     12$,#6          ; NO, LAST SECTOR DONE ??
 2719  023642  103665                          BLO     10$             ; GO 'ROUND AGAIN IF NOT...
 2720  023644                          8$:     EXIT    TST             ;...OTHERWISE, WE'RE ALL DONE.
   (3) 023644  104432                                                                  TRAP    C$EXIT
   (3) 023646  000112                                                                  .WORD   L10062-.
 2721                                  ;
 2722                                  ; SUBROUTINE TO WRITE CHECK A SECTOR AGAINST THE DATA IN 'BUF1".
 2723                                  ;
 2724  023650                          WRITCHK: BGNSEG
   (3) 023650  104404                                                                  TRAP    C$BSEG
 2725  023652  012777  000202  157444          MOV     #CRDY!WRCHK,@RLCS ; SET COMMAND TO WRITE-CHECK...
 2726  023660  013777  003436  157444          MOV     INIMP,@RLMP     ;...128 WORDS...
 2727  023666  013777  003440  157434          MOV     INIDA,@RLDA     ;...FROM CYL X, SECT Y...
 2728  023674  012777  004444  157424          MOV     #BUF1+48.,@RLBA ;...AGAINST BUF1.
 2729  023702  004537  010016                  JSR     R5,BEFORE
 2730  023706  042777  000200  157410          BIC     #CRDY,@RLCS     ; XCT.
 2731  023714  004537  010640                  JSR     R5,WTCRDY       ; WAIT FOR RL DONE.
 2732  023720  123737  024506  003365          CMPB    PSERR,E.CS+1    ; FINAL STATUS RIGHT ??
 2733  023726  001412                          BEQ     1$              ; BR IF SO.
 2734  023730  004537  007510                  JSR     R5,GETERR       ; NO, GET WHATEVER'S THERE.
 2735  023734  000240                          NOP
 2736  023736  013737  024504  003426          MOV     PSETXT,TMP1
 2737  023744                                  DFERR   EM118,ERR20     ; ERROR STATUS WRONG ON WRITE-CHECK
   (5) 023744  104455                                                                  TRAP    C$ERDF
   (6) 023746  000261                                                                  .WORD   177
   (6) 023750  036403                                                                  .WORD   EM118
   (6) 023752  026662                                                                  .WORD   ERR20
 2738  023754                          1$:     ENDSEG
   (3) 023754                                                                  10000$:
   (3) 023754  104405                                                                  TRAP    C$ESEG
 2739  023756  000207                          RTS     PC
 2740  023760                                  ENDTST
   (3) 023760                                                                  L10062:
   (3) 023760  104401                                                                  TRAP    C$ETST
```

```
                                ;
                                ; SUBROUTINE TO SEE IF TLM IS THERE AND PROPERLY CABLED TO RLV12.
                                ; ALSO CHECK THAT CORRECT PROM IS INSTALLED.
                                ; REPORT ERROR AND RETURN WITH "V" SET IF NOT.
                                ;
023762  005000                  TLMOK:  CLR     R0              ; DISABLE ID CHECK.
023764  000401                          SKP1
023766  005200                  TLMPID: INC     R0              ; ENABLE ID CHECK.
023770  013737  003344  003432          MOV     TCSR,GDDAT      ; GET CSR POINTER.
023776  001453                          BEQ     4$              ; ABORT IF IT'S NOT DEFINED.
024000  005037  010626                  CLR     TRPFLG
024004  005077  157334                  CLR     @TCSR           ; TLM ARE YOU REALLY THERE ??
024010  000240                          240
024012  005737  010626                  TST     TRPFLG
024016  001025                          BNE     1$              ; ERROR IF NOT.
024020  012737  004400  003432          MOV     #4400,GDDAT     ; OTHERWISE, ...
024026  017737  157312  003434          MOV     @TCSR,BDDAT     ;...GET INITIAL STATUS.
024034  001423                          BEQ     2$              ; IF ZERO, CABLE ISN'T CONNECTED.
024036  005700                          TST     R0              ; ID VERIFICATION REQUIRED ??
024040  001413                          BEQ     5$              ; WE'RE DONE IF NOT.
024042  004737  024132                  JSR     PC,CLRTLM       ; YES, RESET, ID => R0.
024046  010037  003434                  MOV     R0,BDDAT
024052  013737  003310  003432          MOV     PROMID,GDDAT    ; GET EXPECTED ID.
024060  023737  003432  003434          CMP     GDDAT,BDDAT
024066  001013                          BNE     3$              ; BR IF ID INCORRECT.
024070  000207                  5$:     RTS     PC              ; RETURN, "V" = 0.

024072                          1$:     DFERR   EM100,ERR1      ; TIME-OUT ON TLM ADDRESS.
(5)     024072  104455                                                  TRAP    C$ERDF
(6)     024074  000262                                                  .WORD   178
(6)     024076  034574                                                  .WORD   EM100
(6)     024100  026170                                                  .WORD   ERR1
024102  000411                          BR      4$
024104                          2$:     DFERR   EM100A,ERR2     ; CABLE PROBABLY NOT INSTALLED.
(5)     024104  104455                                                  TRAP    C$ERDF
(6)     024106  000263                                                  .WORD   179
(6)     024110  034630                                                  .WORD   EM100A
(6)     024112  026220                                                  .WORD   ERR2
024114  000404                          BR      4$
024116                          3$:     DFERR   EM100B,ERR2     ; PROM ID INCORRECT.
(5)     024116  104455                                                  TRAP    C$ERDF
(6)     024120  000264                                                  .WORD   180
(6)     024122  034713                                                  .WORD   EM100B
(6)     024124  026220                                                  .WORD   ERR2
024126  000262                  4$:     SEV                     ; SET "ERROR" FLAG...
024130  000207                          RTS     PC              ;...AND RETURN.
                                ;
                                ; SUBROUTINE TO RESET TEST LOOP MODULE.
                                ; RETURN WITH "DRIVE READY" SET, TLM PROM ID IN R0,
                                ; AND TLM CSR ADDRESS IN R5.
                                ; AS LONG AS WE'RE HERE, CLEAR RLBA, RLMP, AND RLBAE AS WELL.
                                ;
024132  013705  003344          CLRTLM: MOV     TCSR,R5         ; DEDICATE R5 TO THE TLM.
024136  012715  000201                  MOV     #1,(R5)         ; RESET...
024142  016500  000006                  MOV     6(R5),R0        ;...PROM ID => R0.
024146  005065  000002                  CLR     2(R5)           ; CLEAR SKGS...
```

```
                                                M 8
002786  024152  005065  000004                        CLR     4(R5)           ;...AND PSUEDO-STATUS REGISTERS.
002787  024156  012715  000002                        MOV     #2,(R5)         ; CLEAR FLAGS ??
002788  024162  012715  000200                        MOV     #200,(R5)       ; SET DRIVE READY.
002789  024166  005077  157134                        CLR     @RLBA
002790  024172  005077  157134                        CLR     @RLMP
002791  024176  023727  003402  000003                CMP     RLTYP,#RLV12X
002792  024204  001002                                BNE     1$
002793  024206  005077  157122                        CLR     @RLBAE
002794  024212  000207                          1$:   RTS     PC
002795                                          ;
002796                                          ; SUBROUTINE TO SET UP BUFFERS AND DISK ADDRESSES
002797                                          ; FOR READING EACH OF THE 6 PSUEDO SECTORS.
002798                                          ; USE HEADER TABLE AND SEARCH ROUTINE IN THE DUMP UTILITY.
002799                                          ; IF SUCCESSFUL, CONTENTS OF SELECTED SECTOR (160. WORDS) ARE IN "BUF1".
002800                                          ; CALL: JSR PC,PSX
002801                                          ;       N               ; SECTOR NUMBER.
002802                                          ;       BVC     XX      ; ON RETURN, "V" IS SET IN SECTOR NOT FOUND.
002803                                          ;       ERROR
002804                                          ; XX:                   ; CONTINUE.
002805                                          ;
002806  024214  017600  000000                  PSX:  MOV     @(SP),R0        ; GET SECTOR NUMBER (1 - 6)...
002807  024220  005300                                DEC     R0              ;...MAKE IT (0 - 5)...
002808  024222  006300                                ASL     R0              ;...SHIFT UP TO AN INDEX.
002809  024224  062716  000002                        ADD     #2,(SP)         ; ADJUST RETURN PC...
002810  024230  000170  024234                        JMP     @1$(R0)         ;...AND DO ONE OF THE FOLLOWING:
002811  024234  024252                          1$:   PS1
002812  024236  024270                                PS2
002813  024240  024322                                PS3
002814  024242  024354                                PS4
002815  024244  024372                                PS5
002816  024246  024442                                PS6
002817  024250  024410                                PS7             ; SOONER OR LATER SOME TURKEY WILL...
002818                                                                  ;...TRY TO ACCESS SECTOR 7.
002819
002820                                                  .ENABL  LSB
002821  024252  112737  000061  036532  PS1:   MOVB    #'1,TSECX       ; TLM SECTOR 1...
002822  024260  012737  000205  003440          MOV     #<1_7.>!5,INIDA ;...IS CYL 1 SECT 5...
002823  024266  000473                          BR      1$              ;...AND IS ERROR FREE.
002824
002825  024270  112737  000062  036532  PS2:   MOVB    #'2,TSECX       ; TLM SECTOR 2...
002826  024276  012737  000206  003440          MOV     #<1_7.>!6,INIDA ;...IS CYL 1, SECT 6...
002827  024304  012737  000210  024506          MOV     #^H<ERR!DCRC>,PSERR
002828  024312  012737  036650  024504          MOV     #EXPDCRC,PSETXT ;...AND HAS A BAD DATA CRC.
002829  024320  000463                          BR      2$
002830
002831  024322  112737  000063  036532  PS3:   MOVB    #'3,TSECX       ; TLM SECTOR 3...
002832  024330  012737  000207  003440          MOV     #<1_7.>!7,INIDA ;...IS CYL 1, SECT 7...
002833  024336  012737  000214  024506          MOV     #^H<ERR!HCRC!OPI>,PSERR
002834  024344  012737  036624  024504          MOV     #EXPHCRC,PSETXT ;...AND HAS A BAD HEADER CRC.
002835  024352  000446                          BR      2$
002836
002837  024354  112737  000064  036532  PS4:   MOVB    #'4,TSECX       ; TLM SECTOR 4...
002838  024362  012737  125252  003440          MOV     #<525_7.>!52,INIDA ;...IS CYL 525, SECT 52...
002839  024370  000432                          BR      1$              ;...AND IS ERROR FREE.
002840
002841  024372  112737  000065  036532  PS5:   MOVB    #'5,TSECX       ; TLM SECTOR 5...
```

```
                                                                                                                    N 8
 2842  024400  012737  135252  003440          MOV     #<565_7.>!52,INIDA ;...IS CYL 565, SECT 52...
 2843  024406  000423                          BR      1$                 ;...AND IS ERROR FREE.
 2844
 2845  024410                          PS7:    PRINTF  #TURKY             ; THERE ISN'T ANY SECTOR 7...
   (7) 024410  012746  024510                                                                          MOV     #TURKY,-(SP)
   (6) 024414  012746  000001                                                                          MOV     #1,-(SP)
   (3) 024420  010600                                                                                  MOV     SP,R0
   (4) 024422  104417                                                                                  TRAP    C$PNTF
   (4) 024424  062706  000004                                                                          ADD     #4,SP
 2846  024430  012737  000006  003314          MOV     #6,SNGLSEC         ;...CHANGE IT TO 6...
 2847  024436  162700  000002                  SUB     #2,R0              ;...ADJUST INDEX AND FALL THRU.
 2848
 2849  024442  112737  000066  036532  PS6:    MOVB    #'6,TSECX          ; TLM SECTOR 6...
 2850  024450  012737  155555  003440          MOV     #<666_7.>!155,INIDA ;...IS CYL 666, SECT 55, HEAD 1...
 2851  024456  005037  024506          1$:     CLR     PSERR
 2852  024462  012737  036670  024504          MOV     #EXPNON,PSETXT
 2853  024470  016037  024572  024570  2$:     MOV     DTABL(R0),DHEAD    ; SET HCRC TO SEARCH FOR.
 2854  024476  004737  025174                  JSR     PC,FINDSEC         ; GO FIND AND READ IT TO BUF1.
 2855  024502  000207                          RTS     PC                 ; RETURN, 'V' = 1 IF SECTOR NOT FOUND.
 2856                                          .DSABL  LSB
 2857
 2858  024504  036670                  PSETXT: EXPNON                     ; EXPECTED ERROR STATUS FOR EACH...
 2859  024506  000000                  PSERR:  0                          ;...IS RETURNED IN THESE LOCATIONS.
 2860
 2861  024510  047045  040445  042523  TURKY:  .ASCIZ  /%N%ASECTOR 7 DOESN'T EXIST, USING 6 INSTEAD%N/
 2862                                          .EVEN
```

```
                        .SBTTL          TLM PROM DUMP UTILITY.
                        ;
                        ;ROUTINE TO DUMP TLM ROM CONTENTS.
                        ;CALLED FROM "INIT" SECTION IF PDSW=1.
                        ;GET UNIT NUMBER, SECTOR NUMBER, AND OUTPUT FORMAT FROM OPERATOR.
                        ;<^C> = ABORT AND RESTART DRS.
                        ;LOAD SUBROUTINE USES "BUF2" AS A 256. WORD RING BUFFER FOR SEARCHING.
                        ;IF FOUND, SECTOR CONTENTS (160. WORDS) ARE THEN COPIED TO "BUF1".
                        ;THE LOAD SUBROUTINE IS ALSO USED IN TLM TESTS 5 AND 6
                        ;TO ESTABLISH EXPECTED DATA PATTERNS.
                        ;
                        ;PSUEDO-DISK (ROM) SECTOR FORMAT:
                        ;       WORDS                   FUNCTION
                        ;       -------                 ----------
                        ;         1 -  17       SECTOR PULSE (12), T3(2), HEADER PREAMBLE (3).
                        ;        18 -  20       DISK ADDRESS (1), ZERO (1), HEADER CRC (1).
                        ;        21 -  24       HEADER POSTAMBLE (1), DATA PREAMBLE (3).
                        ;        25 - 153       DATA (128.), DATA CRC (1).
                        ;       154 - 160       DATA POSTAMBLE (1), AND GAP (6).
                        ;
024566  025324          DMODE:  OCTMOD          ; DUMP MODE (OCTAL OR HEX).
024570  006051          DHEAD:  006051          ; DESIRED HEADER CRC WORD (DEFAULT SECT 1)...
024572  006051          DTABL:  006051  ; 0C29  ;...SELECTED FROM THIS TABLE.
024574  044051                  044051  ; 4829
024576  132051                  132051  ; B429
024600  174001                  174001  ; F801
024602  036400                  036400  ; 3D00
024604  113554                  113554  ; 976C
                        ;
024606  005037  003306  ROMDUMP: CLR     PDSW            ; CLEAR DUMP REQUEST.
024612  005000                  CLR     R0
024614  077001                  SOB     R0,.            ; DELAY...
024616  104005                  EMT+5                   ;...AND CLEAR KBD.
024620  113702  025701  1$:     MOVB    HUNIT,R2        ; GET DEFAULT UNIT.
024624  004737  025476          JSR     PC,KBDOUT       ; SAY "HELLO".
024630  025561                  Q0
024632  004737  025432          JSR     PC,KBDIN        ; GET A UNIT NUMBER.
024636  113700  004364          MOVB    BUF1,R0
024642  001407                  BEQ     2$              ; IF NONE, USE DEFAULT (LAST).
024644  120027  000060          CMPB    R0,#'0
024650  103763                  BLO     1$              ; BR IF INVALID UNIT.
024652  120027  000067          CMPB    R0,#'7
024656  101360                  BHI     1$              ;       DITTO
024660  110002                  MOVB    R0,R2           ; SAVE NEW UNIT FOR A MOMENT.
024662  110201          2$:     MOVB    R2,R1
024664  013705  003320          MOV     TC$0,R5         ; BASE CSR => R5.
024670  042701  177770          BIC     #^C7,R1         ; STRIP UNIT NUM...
024674  001403                  BEQ     3$              ;...AND BR IF ZERO.
024676  062705  000010  22$:     ADD     #10,R5          ; ELSE, ADJUST UP TO SELECTED UNIT.
024702  077103                  SOB     R1,22$
                        ;
024704  110237  025701  3$:     MOVB    R2,HUNIT        ; OK, SET UNIT IN TEXT.
024710  012737  024726  000004  MOV     #4$,ERRVEC      ; SET A TRAP CATCHER.
024716  012715  000001          MOV     #1,(R5)         ; RESET TLM...
024722  000240                  240
024724  000410                  BR      5$              ;...AND PROCEED IF IT DIDN'T TRAP.
```

```
2920  024726  022626                    4$:   CMP     (SP)+,(SP)+
2921  024730  113737  025701  025536          MOVB    HUNIT,NOANS+6
2922  024736  004737  025476                  JSR     PC,KBDOUT       ; SELECTED UNIT DOESN'T ANSWER.
2923  024742  025530                          NOANS
2924  024744  000720                          BR      ROMDUMP
2925
2926  024746  004737  025476            5$:   JSR     PC,KBDOUT       ; GET A SECTOR NUMBER.
2927  024752  025612                          Q1
2928  024754  004737  025432                  JSR     PC,KBDIN
2929  024760  113700  004364                  MOVB    BUF1,R0
2930  024764  001416                          BEQ     6$              ; IF NONE, REDO LAST SECTOR.
2931  024766  120027  000060                  CMPB    R0,#'0
2932  024772  103765                          BLO     5$
2933  024774  120027  000067                  CMPB    R0,#'7
2934  025000  101362                          BHI     5$
2935  025002  110037  025714                  MOVB    R0,HSECT        ; SET IN HEADER STRING.
2936  025006  042700  177770                  BIC     #^C7,R0
2937  025012  006300                          ASL     R0              ; SHIFT UP TO WORD OFFSET...
2938  025014  016037  024570  024570          MOV     DHEAD(R0),DHEAD ;...AND SET HEADER TO SEARCH FOR.
2939
2940  025022  004737  025476            6$:   JSR     PC,KBDOUT       ; GET AN OUTPUT MODE.
2941  025026  025637                          Q2
2942  025030  004737  025432                  JSR     PC,KBDIN
2943  025034  113700  004364                  MOVB    BUF1,R0
2944  025040  001411                          BEQ     7$              ; IF NONE, USE CURRENT FORMAT.
2945  025042  012737  025324  024566          MOV     #OCTMOD,DMODE   ; ASSUME OCTAL...
2946  025050  120027  000131                  CMPB    R0,#'Y
2947  025054  001003                          BNE     7$
2948  025056  012737  025350  024566          MOV     #HEXMOD,DMODE   ;...CHANGE TO HEX IF REQUIRED.
2949  025064                            7$:
2950  025064  016537  000006  003424  DOIT:   MOV     6(R5),TMP0      ; GET PROM ID.
2951  025072  004737  025476                  JSR     PC,KBDOUT       ; SHOW UNIT AND SECTOR...
2952  025076  025670                          HEADR
2953  025100  012701  003424                  MOV     #TMP0,R1        ;...AND PROM ID...
2954  025104  004777  177456                  JSR     PC,@DMODE       ;...IN THE CURRENT CONTEXT.
2955  025110  104007                          EMT+7                   ; <CRLF>
2956  025112  004737  025174                  JSR     PC,FINDSEC      ; SEARCH FOR AND READ SECTOR.
2957  025116  102004                          BVC     1$              ; BR IF SECTOR FOUND.
2958  025120  004737  025476                  JSR     PC,KBDOUT
2959  025124  025731                          CANTFIND
2960  025126  000627                          BR      ROMDUMP
2961
2962  025130  012701  004364            1$:   MOV     #BUF1,R1        ; NOW FOUND SECTOR IS IN BUF1.
2963  025134  012702  000240                  MOV     #160.,R2        ; DUMP 160. WORDS AT 8 PER LINE.
2964  025140  104007                          EMT+7                   ; <CRLF>
2965  025142  004737  025476            2$:   JSR     PC,KBDOUT
2966  025146  025756                          TAB
2967  025150  004777  177412                  JSR     PC,@DMODE       ; PRINT FROM (R1) IN OCTAL OR HEX.
2968  025154  005302                          DEC     R2
2969  025156  032702  000007                  BIT     #7,R2
2970  025162  001367                          BNE     2$              ; LOOP 'TIL LINE DONE...
2971  025164  104007                          EMT+7                   ; <CRLF>
2972  025166  005702                          TST     R2
2973  025170  001364                          BNE     2$              ;...AND LOOP 'TIL ALL DONE.
2974  025172  000605                          BR      ROMDUMP
2975                                    ;
```

```
                                                                D 9
2976
2977                            ; SEARCH FOR SELECTED SECTOR AND LOAD 160. WORD BLOCK TO BUF1.
2978                            ; IF UNSUCCESSFUL, RETURN TO CALLER WITH "V" BIT SET.
2979                            ;
2980 025174  012715  000001    FINDSEC: MOV     #1,(R5)         ; RESET TLM...
2981 025200  012715  000034             MOV     #34,(R5)        ;...AND SET PROM READ MODE.
2982 025204  005046                     CLR     -(SP)           ; CLEAR "FOUND" FLAG.
2983 025206  005000                     CLR     R0              ; SET LOOP CONTROL...
2984 025210  012701  006364             MOV     #BUFEND,R1      ; NOW, FILL BUF2 FROM TOP DOWN.
2985 025214  020127  005364     1$:     CMP     R1,#BUF2        ; IN LOOP -- RESET RING POINTER...
2986 025220  001002                     BNE     2$              ;...WHEN IT REACHES THE BOTTOM.
2987 025222  012701  006364             MOV     #BUFEND,R1
2988 025226  005265  000006     2$:     INC     6(R5)           ; INCR ROM ADDRESS...
2989 025232  016541  000006             MOV     6(R5),-(R1)     ;...AND GET A WORD.
2990 025236  021137  024570             CMP     (R1),DHEAD      ; IS THIS THE DESIRED HCRC WORD ??
2991 025242  001003                     BNE     3$              ; BR IF NOT.
2992 025244  005216                     INC     (SP)            ; YES, SET "FOUND" FLAG...
2993 025246  012700  000024             MOV     #20.,R0         ;...AND SET TO READ 19 MORE.
2994 025252  077020             3$:     SOB     R0,1$           ; LOOP 'TIL DONE...
2995 025254  012715  000001             MOV     #1,(R5)         ;...THEN, RESET TLM...
2996 025260  012715  000200             MOV     #200,(R5)       ;...AND RESTORE DRIVE READY.
2997 025264  005726                     TST     (SP)+           ; SECTOR FOUND ??
2998 025266  001002                     BNE     4$              ; BR IF SO.
2999 025270  000262                     SEV                     ; NO, SET "V" BIT...
3000 025272  000207                     RTS     PC              ;...AND RETURN.
3001
3002 025274  012702  004364     4$:     MOV     #BUF1,R2        ; NOW COPY FOUND SECTOR TO BUF1.
3003 025300  012703  000240             MOV     #160.,R3
3004 025304  020127  006364     5$:     CMP     R1,#BUFEND      ; IN LOOP -- RESET RING POINTER...
3005 025310  001002                     BNE     6$              ;...WHEN IT REACHES THE TOP.
3006 025312  012701  005364             MOV     #BUF2,R1
3007 025316  012122             6$:     MOV     (R1)+,(R2)+     ; BUF2 => BUF1.
3008 025320  077307                     SOB     R3,5$
3009 025322  000207                     RTS     PC              ; RETURN, "V" IS CLEAR.
3010                            ;
3011                            ; SUBROUTINES TO DUMP DATA IN OCTAL OR HEX.
3012                            ;
3012 025324                     OCTMOD: PRINTF  #OCTTXT,(R1)+
  (8) 025324  012146                                                    MOV     (R1)+,-(SP)
  (7) 025326  012746  025770                                            MOV     #OCTTXT,-(SP)
  (6) 025332  012746  000002                                            MOV     #2,-(SP)
  (3) 025336  010600                                                    MOV     SP,R0
  (4) 025340  104417                                                    TRAP    C$PNTF
  (4) 025342  062706  000006                                            ADD     #6,SP
3013 025346  000207                     RTS     PC
3014
3015 025350  010546             HEXMOD: MOV     R5,-(SP)        ; SAVE TLM POINTER.
3016 025352  012700  025763     1$:     MOV     #HEXTXT,R0      ; SET HEX TEXT POINTER.
3017 025356  012104                     MOV     (R1)+,R4        ; NEXT WORD => R4.
3018 025360  005005             2$:     CLR     R5              ;...AND PARSE EACH NIBBLE IN R5.
3019 025362  012703  000004             MOV     #4,R3
3020 025366  006304             3$:      ASL    R4
3021 025370  006105                       ROL   R5              ; NIBBLE => R5<3:0>
3022 025372  077303                        SOB  R3,3$
3023 025374  062705  000060             ADD     #'0,R5          ; CONVERT TO ASCII 0 TO 9...
3024 025400  120527  000071             CMPB    R5,#'9
3025 025404  003402                     BLE     4$
```

```
 3026  025406  062705  000007          ADD     #7,R5           ;...AND A(10) TO F(15)...
 3027  025412  110520                  4$:     MOVB    R5,(R0)+        ;...AND STUFF IT.
 3028  025414  105710                          TSTB    (R0)
 3029  025416  001360                          BNE     2$              ; LOOP 'TIL 4 CHARS ASSEMBLED...
 3030  025420  004737  025476                  JSR     PC,KBDOUT       ;...THEN DUMP IT.
 3031  025424  025763                          HEXTXT
 3032  025426  012605                          MOV     (SP)+,R5        ; RESTORE R5.
 3033  025430  000207                          RTS     PC
 3034
 3035  025432  012701  004364          KBDIN:  MOV     #BUF1,R1        ; TEMP ASCII BUFFER.
 3036  025436  104005                  1$:     EMT+5                   ; CHAR => R0.
 3037  025440  110011                          MOVB    R0,(R1)
 3038  025442  001775                          BEQ     1$
 3039  025444  120027  000003                  CMPB    R0,#3
 3040  025450  001411                          BEQ     3$              ; EXIT ON <^C>.
 3041  025452  120027  000015                  CMPB    R0,#15
 3042  025456  001403                          BEQ     2$              ; BR ON TERMINATOR.
 3043  025460  104004                          EMT+4                   ; OTHERWISE, ECHO...
 3044  025462  005201                          INC     R1              ;...BUMP POINTER...
 3045  025464  000764                          BR      1$              ;...AND LOOP.
 3046  025466  105011                  2$:     CLRB    (R1)            ; TERMINATE INPUT STRING...
 3047  025470  104007                          EMT+7                   ;...ECHO CRLF...
 3048  025472  000207                          RTS     PC              ;...AND RETURN TO CALLER.
 3049  025474                          3$:     DOCLN                   ; EXIT ON <^C>.
   (3) 025474  104444                                                          TRAP    C$DCLN
 3050
 3051  025476                          KBDOUT: PRINTF  #PURE,@(SP)
   (8) 025476  017646  000000                                                  MOV     @(SP),-(SP)
   (7) 025502  012746  025774                                                  MOV     #PURE,-(SP)
   (6) 025506  012746  000002                                                  MOV     #2,-(SP)
   (5) 025512  010600                                                          MOV     SP,R0
   (4) 025514  104417                                                          TRAP    C$PNTF
   (4) 025516  062706  000006                                                  ADD     #6,SP
 3052  025522  062716  000002                  ADD     #2,(SP)
 3053  025526  000207                          RTS     PC
 3054
 3055  025530  020040  046124  035115  NOANS:  .ASCIZ  /  TLM:0 DOESN'T ANSWER/<15><12>
 3056  025561     015     012          Q0:     .ASCII  <15><12>
 3057  025563     040  052040  046514          .ASCIZ  /  TLM UNIT (0 TO 7)    ? /
 3058  025612  020040  042523  052103  Q1:     .ASCIZ  /  SECTOR (1 TO 6)      ? /
 3059  025637     040  044040  054105  Q2:     .ASCIZ  /  HEX OUTPUT (Y OR N)  ? /
 3060  025670  005015  052411  044516  HEADR:  .ASCII  <15><12><11>/UNIT: /
 3061  025701     060  020040  042523  HUNIT:  .ASCII  /0  SECTOR: /
 3062  025714  020061  050040  047522  HSECT:  .ASCIZ  /1  PROM ID: /
 3063  025731     011  040503  023516  CANTFIND: .ASCIZ <11>/CAN'T FIND HEADER/<15><12>
 3064  025756  000011                  TAB:    .ASCIZ  <11>
 3065  025760  005015     000          CRLF:   .ASCIZ  <15><12>
 3066  025763     101  041502  000104  HEXTXT: .ASCIZ  /ABCD/
 3067  025770  047445  000066          OCTTXT: .ASCIZ  /%06/
 3068  025774  052045     000          PURE:   .ASCIZ  /%T/
 3069  026000                          .EVEN
```

```
                        .SBTTL
                        .SBTTL  SUPERVISOR DISPATCH TABLE.
                        ;
                        ; SUBROUTINE TO ADJUST THE NUMBER OF TESTS TO RUN IAW CONTROLLER TYPE.
                        ;
026000  013700  003402  ADJTN: MOV     RLTYP,R0        ; GET RL TYPE.
026004  001414                 BEQ     2$              ; BR IF RL11.
026006  020027  000002         CMP     R0,#RLV12
026012  103406                 BLO     1$              ; BR IF RLV11.
026014  005737  003344         TST     TCSR
026020  001403                 BEQ     1$              ; BR IF TLM BYPASSED.
026022  012700  000043         MOV     #35.,R0         ; RLV12 WITH TLM RUNS ALL.
026026  000402                 SKP2
026030  012700  000035 1$:     MOV     #29.,R0         ; RLV12 OR RLV11 RUN THRU 29.
026034  000402                 SKP2
026036  012700  000026 2$:     MOV     #22.,R0         ; RL11 RUN THRU 22.
026042  010037  026050         MOV     R0,L$DISPATCH-2 ;...AND SET LAST TEST NUMBER.
026046  000207                 RTS     PC
                        ;
026050                  DISPATCH T$TESTNUM
(4)     026050  000043                                          .WORD   35
(3)     026052                                          L$DISPATCH::
(6)     026052  012114                                          .WORD   T1
(6)     026054  012200                                          .WORD   T2
(6)     026056  012264                                          .WORD   T3
(6)     026060  012350                                          .WORD   T4
(6)     026062  012434                                          .WORD   T5
(6)     026064  012526                                          .WORD   T6
(6)     026066  013020                                          .WORD   T7
(6)     026070  013140                                          .WORD   T8
(6)     026072  013244                                          .WORD   T9
(6)     026074  013332                                          .WORD   T10
(6)     026076  013436                                          .WORD   T11
(6)     026100  013664                                          .WORD   T12
(6)     026102  014050                                          .WORD   T13
(6)     026104  014216                                          .WORD   T14
(6)     026106  014410                                          .WORD   T15
(6)     026110  014612                                          .WORD   T16
(6)     026112  015036                                          .WORD   T17
(6)     026114  015262                                          .WORD   T18
(6)     026116  015532                                          .WORD   T19
(6)     026120  015746                                          .WORD   T20
(6)     026122  016160                                          .WORD   T21
(6)     026124  016246                                          .WORD   T22
(6)     026126  016414                                          .WORD   T23
(6)     026130  016550                                          .WORD   T24
(6)     026132  016634                                          .WORD   T25
(6)     026134  016762                                          .WORD   T26
(6)     026136  017662                                          .WORD   T27
(6)     026140  020306                                          .WORD   T28
(6)     026142  020502                                          .WORD   T29
(6)     026144  021276                                          .WORD   T30
(6)     026146  021650                                          .WORD   T31
(6)     026150  022054                                          .WORD   T32
(6)     026152  022222                                          .WORD   T33
(6)     026154  022502                                          .WORD   T34
```

```
   (6)  026156  023364                                          .WORD   T35
```

```
                                                        H 9
3092
3093
3094
3095
3096  026160
  (3) 026160
3097  026160  000137  027306          NOSIG:  JMP     CKERLT          ; SOME HAVE NO SIGNATURE.
3098                                                                                ERRSIG::
3099  026164  000137  027100          ERR0:   JMP     ALLREGS         ; ALL REGISTERS.
3100
3101  026170
  (8) 026170  013746  003432          ERR1:   PRINTB  #FRMT2,GDDAT    ; BUS-TIMEOUT REG ADDR.
  (7) 026174  012746  027501                                                    MOV     GDDAT,-(SP)
  (6) 026200  012746  000002                                                    MOV     #FRMT2,-(SP)
  (3) 026204  010600                                                            MOV     #2,-(SP)
  (4) 026206  104414                                                            MOV     SP,R0
  (4) 026210  062706  000006                                                    TRAP    C$PNTB
3102  026214  000137  027306                                                    ADD     #6,SP
3103                                          JMP     CKERLT
3104  026220
  (9) 026220  013746  003434          ERR2:   PRINTB  #FRMT4,GDDAT,BDDAT      ; EXP'D VS REC'D.
  (8) 026224  013746  003432                                                    MOV     BDDAT,-(SP)
  (7) 026230  012746  027602                                                    MOV     GDDAT,-(SP)
  (6) 026234  012746  000003                                                    MOV     #FRMT4,-(SP)
  (3) 026240  010600                                                            MOV     #3,-(SP)
  (4) 026242  104414                                                            MOV     SP,R0
  (4) 026244  062706  000010                                                    TRAP    C$PNTB
3105  026250  000137  027306                                                    ADD     #10,SP
3106                                          JMP     CKERLT
3107  026254
  (8) 026254  013746  003434          ERR3:   PRINTB  #FRMT5,BDDAT    ; MAINT SEQ FAILED STATE.
  (7) 026260  012746  027636                                                    MOV     BDDAT,-(SP)
  (6) 026264  012746  000002                                                    MOV     #FRMT5,-(SP)
  (3) 026270  010600                                                            MOV     #2,-(SP)
  (4) 026272  104414                                                            MOV     SP,R0
  (4) 026274  062706  000006                                                    TRAP    C$PNTB
3108  026300  000137  027100                                                    ADD     #6,SP
3109                                          JMP     ALLREGS
3110  026304
  (9) 026304  013746  003434          ERR4:   PRINTB  #FRMT4,GDDAT,BDDAT      ; EXP'D VS REC'D...
  (8) 026310  013746  003432                                                    MOV     BDDAT,-(SP)
  (7) 026314  012746  027602                                                    MOV     GDDAT,-(SP)
  (6) 026320  012746  000003                                                    MOV     #FRMT4,-(SP)
  (3) 026324  010600                                                            MOV     #3,-(SP)
  (4) 026326  104414                                                            MOV     SP,R0
  (4) 026330  062706  000010                                                    TRAP    C$PNTB
3111  026334  000137  027100                                                    ADD     #10,SP
3112                                          JMP     ALLREGS                 ;...AND REGISTERS.
3113  026340
  (6) 026340  012746  034404          ERR6:   PRINTB  #FRMT3,#EM99    ; CS ERRORS.
  (7) 026344  012746  027517                                                    MOV     #EM99,-(SP)
  (6) 026350  012746  000002                                                    MOV     #FRMT3,-(SP)
  (3) 026354  010600                                                            MOV     #2,-(SP)
  (4) 026356  104414                                                            MOV     SP,R0
  (4) 026360  062706  000006                                                    TRAP    C$PNTB
3114  026364  000137  027100                                                    ADD     #6,SP
                                          JMP     ALLREGS
```

```
                                                                    I 9
3115
3116  026370
  (8) 026370  013746  003424      ERR7:   PRINTB  #FRMT6,TMP0       ; CPU PRIORITY.
  (7) 026374  012746  027670                                                  MOV     TMP0,-(SP)
  (6) 026400  012746  000002                                                  MOV     #FRMT6,-(SP)
  (3) 026404  010600                                                          MOV     #2,-(SP)
  (4) 026406  104414                                                          MOV     SP,R0
  (4) 026410  062706  000006                                                  TRAP    C$PNTB
3117  026414  000137  027306                                                  ADD     #6,SP
3118                                        JMP     CKERLT
3119  026420
  (8) 026420  012746  034404      ERR10:  PRINTB  #FRMT3A,#EM99   ; EXPECTED OPI ERRORS.
  (7) 026424  012746  027541                                                  MOV     #EM99,-(SP)
  (6) 026430  012746  000002                                                  MOV     #FRMT3A,-(SP)
  (3) 026434  010600                                                          MOV     #2,-(SP)
  (4) 026436  104414                                                          MOV     SP,R0
  (4) 026440  062706  000006                                                  TRAP    C$PNTB
3120  026444  000137  027100                                                  ADD     #6,SP
3121                                        JMP     ALLREGS
3122  026450
  (8) 026450  013746  003434      ERR11:  PRINTB  #FRMT10,BDDAT   ; OPI TIMING ERROR.
  (7) 026454  012746  027723                                                  MOV     BDDAT,-(SP)
  (6) 026460  012746  000002                                                  MOV     #FRMT10,-(SP)
  (3) 026464  010600                                                          MOV     #2,-(SP)
  (4) 026466  104414                                                          MOV     SP,R0
  (4) 026470  062706  000006                                                  TRAP    C$PNTB
3123  026474  000137  027306                                                  ADD     #6,SP
3124                                        JMP     CKERLT
3125  026500  004737  026610      ERR12:  JSR     PC,PRTPA        ; STAT WRONG ON NXM (ABCVE 32K)...
3126  026504                              PRINTB  #FRMT21,#EM99   ;...OR BBS7/NXM FAILURE.
  (8) 026504  012746  034404                                                  MOV     #EM99,-(SP)
  (7) 026510  012746  030241                                                  MOV     #FRMT21,-(SP)
  (6) 026514  012746  000002                                                  MOV     #2,-(SP)
  (3) 026520  010600                                                          MOV     SP,R0
  (4) 026522  104414                                                          TRAP    C$PNTB
  (4) 026524  062706  000006                                                  ADD     #6,SP
3127  026530  000137  027100                JMP     ALLREGS
3128
3129  026534  004737  026610      ERR13:  JSR     PC,PRTPA        ; UNEX STAT ERRORS (ABOVE 32K).
3130  026540  000137  026340              JMP     ERR6
3131
3132  026544  004737  026610      ERR14:  JSR     PC,PRTPA        ; DATA WRONG (ABOVE 32K).
3133  026550                              PRINTB  #FRMT22,TMP0,GDDAT,BDDAT ; EXPD VS RECD.
 (10) 026550  013746  003434                                                  MOV     BDDAT,-(SP)
  (9) 026554  013746  003432                                                  MOV     GDDAT,-(SP)
  (8) 026560  013746  003424                                                  MOV     TMP0,-(SP)
  (7) 026564  012746  030276                                                  MOV     #FRMT22,-(SP)
  (6) 026570  012746  000004                                                  MOV     #4,-(SP)
  (3) 026574  010600                                                          MOV     SP,R0
  (4) 026576  104414                                                          TRAP    C$PNTB
  (4) 026600  062706  000012                                                  ADD     #12,SP
3134  026604  000137  027100                JMP     ALLREGS
3135
3136                                    ;
3137                                    ; PACK AND PRINT 22 BIT PHYSICAL ADDRESS FROM TMP1,,TMP2.
3138  026610  005737  003430      PRTPA:  TST     TMP2            ; BIT 15 ON ??
```

```
                                                                J 9
3139  026614  100003
3140  026616  062737  100000  003430          BPL     1$                      ; NO, "C" = 0,
3141  026624  006137  003426                  ADD     #BIT15,TMP2             ; YES, "C" = 1,
3142  026630                          1$:     ROL     TMP1                    ; TMP1 = ADDR<21:15>...
    (9) 026630  013746  003430                PRINTB  #FRMT20,TMP1,TMP2 ; TMP2 = ADDR<14:00>.
    (8) 026634  013746  003426                                                      MOV     TMP2,-(SP)
    (7) 026640  012746  030201                                                      MOV     TMP1,-(SP)
    (6) 026644  012746  000003                                                      MOV     #FRMT20,-(SP)
    (3) 026650  010600                                                              MOV     #3,-(SP)
    (4) 026652  104414                                                              MOV     SP,R0
    (4) 026654  062706  000010                                                      TRAP    C$PNTB
                                                                                    ADD     #10,SP
3143  026660  000207                          RTS     PC
3144
3145                                    ;
3146                                    ; TLM UNIQUE ERROR SIGNATURES.
3147  026662  004737  027042            ERR20:  JSR     PC,E2022                ; BAD STATUS AFTER READ,READNH,WRTCHK...
3148  026666                            ERR20A:                                 ;...OR AFTER SEEK,GET STATUS,WRITE.
3149  026666                                    PRINTX  #FRMT26,TMP1,#EM99
    (9) 026666  012746  034404                                                      MOV     #EM99,-(SP)
    (8) 026672  013746  003426                                                      MOV     TMP1,-(SP)
    (7) 026676  012746  030410                                                      MOV     #FRMT26,-(SP)
    (6) 026702  012746  000003                                                      MOV     #3,-(SP)
    (3) 026706  010600                                                              MOV     SP,R0
    (4) 026710  104415                                                              TRAP    C$PNTX
    (4) 026712  062706  000010                                                      ADD     #10,SP
3150  026716                                    PRINTX  #FRMT27                 ; CRLF.
    (7) 026716  012746  030431                                                      MOV     #FRMT27,-(SP)
    (6) 026722  012746  000001                                                      MOV     #1,-(SP)
    (3) 026726  010600                                                              MOV     SP,R0
    (4) 026730  104415                                                              TRAP    C$PNTX
    (4) 026732  062706  000004                                                      ADD     #4,SP
3151  026736  000137  027100                    JMP     ALLREGS
3152
3153  026742  004737  027042    ERR21:  JSR     PC,E2022                ; READ HEADER FAILS.
3154  026746  000137  027306            JMP     CKERLT
3155
3156  026752  004737  027042    ERR22:  JSR     PC,E2022                ; DATA FROM TLM WRONG.
3157  026756                            PRINTX  #FRMT14,TMP1,TMP0,GDDAT,BDDAT
   (11) 026756  013746  003434                                                      MOV     BDDAT,-(SP)
   (10) 026762  013746  003432                                                      MOV     GDDAT,-(SP)
    (9) 026766  013746  003424                                                      MOV     TMP0,-(SP)
    (8) 026772  013746  003426                                                      MOV     TMP1,-(SP)
    (7) 026776  012746  030031                                                      MOV     #FRMT14,-(SP)
    (6) 027002  012746  000005                                                      MOV     #5,-(SP)
    (3) 027006  010600                                                              MOV     SP,R0
    (4) 027010  104415                                                              TRAP    C$PNTX
    (4) 027012  062706  000014                                                      ADD     #14,SP
3158  027016                                    PRINTX  #FRMT27                 ; CRLF
    (7) 027016  012746  030431                                                      MOV     #FRMT27,-(SP)
    (6) 027022  012746  000001                                                      MOV     #1,-(SP)
    (3) 027026  010600                                                              MOV     SP,R0
    (4) 027030  104415                                                              TRAP    C$PNTX
    (4) 027032  062706  000004                                                      ADD     #4,SP
3159  027036  000137  027100                    JMP     ALLREGS
3160
3161  027042                            E2022:  PRINTX  #FRMT25,#TSEC,TMP2,#TPEAK
```

```
(10) 027042 012746 036602                                                       MOV     #TPEAK,-(SP)
 (9) 027046 013746 003430                                                       MOV     TMP2,-(SP)
 (8) 027052 012746 036523                                                       MOV     #TSEC,-(SP)
 (7) 027056 012746 030401                                                       MOV     #FRMT25,-(SP)
 (6) 027062 012746 000004                                                       MOV     #4,-(SP)
 (3) 027066 010600                                                              MOV     SP,R0
 (4) 027070 104415                                                              TRAP    C$PNTX
 (4) 027072 062706 000012                                                       ADD     #12,SP
3162 027076 000207                      RTS     PC
3163                                    ;
3164                                    ; DISPLAY ALL REGISTERS BEFORE AND AFTER ERROR (IF REQ'D),
3165                                    ; CHECK IF ERROR LIMIT EXCEEDED AND EXIT ACCORDINGLY.
3166                                    ;
3167 027100 005000                      ALLREGS: CLR    R0                      ; DISABLE...
3168 027102 023727 003402 000003                CMP     RLTYP,#RLV12X
3169 027110 001002                              BNE     1$
3170 027112 012700 000045                       MOV     #'%,R0                  ;...OR ENABLE THE EXTRA REGISTER.
3171 027116 110037 027425               1$:     MOVB    R0,FRMT1A                       ; ENABLE/DISABLE BAE TEXT.
3172 027122 110037 027467                       MOVB    R0,FRMT1C
3173 027126                                     PRINTB  #FRMT1                  ; HEADER.
 (7) 027126 012746 027366                                                       MOV     #FRMT1,-(SP)
 (6) 027132 012746 000001                                                       MOV     #1,-(SP)
 (3) 027136 010600                                                              MOV     SP,R0
 (4) 027140 104414                                                              TRAP    C$PNTB
 (4) 027142 062706 000004                                                       ADD     #4,SP
3174 027146                             PRINTB  #FRMT1B,#BEREG,B.CS,B.BA,B.DA,B.MP,B.BAE
(13) 027146 013746 003360                                                       MOV     B.BAE,-(SP)
(12) 027152 013746 003356                                                       MOV     B.MP,-(SP)
(11) 027156 013746 003354                                                       MOV     B.DA,-(SP)
(10) 027162 013746 003352                                                       MOV     B.BA,-(SP)
 (9) 027166 013746 003350                                                       MOV     B.CS,-(SP)
 (8) 027172 012746 030434                                                       MOV     #BEREG,-(SP)
 (7) 027176 012746 027436                                                       MOV     #FRMT1B,-(SP)
 (6) 027202 012746 000007                                                       MOV     #7,-(SP)
 (3) 027206 010600                                                              MOV     SP,R0
 (4) 027210 104414                                                              TRAP    C$PNTB
 (4) 027212 062706 000020                                                       ADD     #20,SP
3175 027216                             PRINTB  #FRMT1B,#AFREG,E.CS,E.BA,E.DA,E.MP,E.BAE
(13) 027216 013746 003376                                                       MOV     E.BAE,-(SP)
(12) 027222 013746 003372                                                       MOV     E.MP,-(SP)
(11) 027226 013746 003370                                                       MOV     E.DA,-(SP)
(10) 027232 013746 003366                                                       MOV     E.BA,-(SP)
 (9) 027236 013746 003364                                                       MOV     E.CS,-(SP)
 (8) 027242 012746 030455                                                       MOV     #AFREG,-(SP)
 (7) 027246 012746 027436                                                       MOV     #FRMT1B,-(SP)
 (6) 027252 012746 000007                                                       MOV     #7,-(SP)
 (3) 027256 010600                                                              MOV     SP,R0
 (4) 027260 104414                                                              TRAP    C$PNTB
 (4) 027262 062706 000020                                                       ADD     #20,SP
3176 027266                             PRINTB  #FRMT1D
 (7) 027266 012746 027476                                                       MOV     #FRMT1D,-(SP)
 (6) 027272 012746 000001                                                       MOV     #1,-(SP)
 (3) 027276 010600                                                              MOV     SP,R0
 (4) 027300 104414                                                              TRAP    C$PNTB
 (4) 027302 062706 000004                                                       ADD     #4,SP
3177                                            ; FALL THRU...
```

```
                                                                    L 9
3178                                    ;
3179                                    ;...AND CHECK ERROR LIMIT (IF ANY).
3180                                    ;
3181  027306                            CKERLT: INLOOP
  (3) 027306  104420                                                        TRAP    C$INLP
3182  027310                                    BCOMPLETE 3$              ; RETURN IF LOOPING...
  (2) 027310  103425                                                        BCS     3$
3183  027312  005737  003304                    TST     ERRLMT
3184  027316  001422                            BEQ     3$                ;...OR NO ERROR LIMIT SET.
3185  027320  005277  154142                    INC     @ERPOINT          ;COUNT THE UNIT ERROR DETECTED
3186  027324  027737  154136  003304            CMP     @ERPOINT,ERRLMT   ;REACHED THE ERROR LIMIT?
3187  027332  002414                            BLT     3$                ;NO, RETURN
3188  027334                                    PRINTF  #FRMT23           ; "ERROR LIMIT EXCEEDED"
  (7) 027334  012746  030350                                                MOV     #FRMT23,-(SP)
  (6) 027340  012746  000001                                                MOV     #1,-(SP)
  (3) 027344  010600                                                        MOV     SP,R0
  (4) 027346  104417                                                        TRAP    C$PNTF
  (4) 027350  062706  000004                                                ADD     #4,SP
3189  027354                            2$:     DODU    UNITST            ;DROP THE UNIT...
  (3) 027354  013700  003322                                                MOV     UNITST,R0
  (3) 027360  104451                                                        TRAP    C$DODU
3190  027362                                    DOCLN                     ;...AND ABORT.
  (3) 027362  104444                                                        TRAP    C$DCLN
3191  027364                            3$:     ENDMSG                    ; OR RETURN TO CALLER.
  (3) 027364                                                        L10063:
  (3) 027364  104423                                                        TRAP    C$MSG
```

```
                                        ;
                                        ; FORMATTED ASCII TEXT.
                                        ;
                                FRMT1:  .ASCII  /%AREGISTERS     CS      BA      DA      MP/
                                FRMT1A: .ASCIZ  /%A          BAE/
                                FRMT1B: .ASCII  /%N%T%O6%S2%O6%S2%O6%S2%O6/
                                FRMT1C: .ASCIZ  /%S2%O6/
                                FRMT1D: .ASCIZ  /%N/
                                FRMT2:  .ASCIZ  /%AADDR: %O6%N/
                                FRMT3:  .ASCIZ  /%AERRORS SET:%T%N/
                                FRMT3A: .ASCIZ  /%AEXP'D: ERR HNF OPI  REC'D:%T%N/
                                FRMT4:  .ASCIZ  /%AEXP'D: %O6%A REC'D: %O6%N/
                                FRMT5:  .ASCIZ  /%AAT INTERNAL STATE %D2%N/
                                FRMT6:  .ASCIZ  /%AAT PROCESSOR LEVEL %O1%N/
                                FRMT10: .ASCII  /%AOPI FLAG RECEIVED AT%D4%A MSEC.%N/
                                        .ASCIZ  /%AEXPECTED FROM 155 TO 650 MSEC.%N/
                                FRMT14: .ASCIZ  /%N%AWORD: %O3%A  ADDR: %O6%A  EXP'D: %O6%A  REC'D: %O6/
                                FRMT15: .ASCIZ  /%N%D3%A WORDS BAD OUT OF 255 WORDS TRANSFERRED%N/
                                FRMT20: .ASCIZ  /%AAT PHYSICAL ADDRESS: %O3%O5%N/
                                FRMT21: .ASCIZ  /%AEXP'D: ERR NXM  REC'D:%T%N/
                                FRMT22: .ASCIZ  /%AWORD: %O3%A  EXP'D: %O6%A  REC'D: %O6%N/
                                FRMT23: .ASCIZ  /%AERROR LIMIT EXCEEDED%N/

                                FRMT25: .ASCIZ  /%T%T%T/
                                FRMT26: .ASCIZ  /%N%T%A  REC'D:%T/
                                FRMT27: .ASCIZ  /%N/
                                        .EVEN
                                        ;
                                        ; PURE (UNFORMATTED) ASCII TEXT.
                                        ;
                                BEREG:  .ASCIZ  /BEFORE COMMAND: /
                                AFREG:  .ASCIZ  /AFTER COMMAND:  /
                                CRTIM:  .ASCIZ  /CONTROLLER TIMED OUT/
                                DRTIM:  .ASCIZ  /DRIVE READY TIMED OUT/
                                CERR:   .ASCIZ  / ERR/
                                DEMES:  .ASCIZ  / DRV/
                                NXMMES: .ASCIZ  / NXM/
                                PARMES: .ASCIZ  / PAR/
                                DLTMES: .ASCIZ  / DLT/
                                HNFMES: .ASCIZ  / HNF/
                                DCRCMES: .ASCIZ / DCRC/
                                HCRCMES:.ASCIZ  / HCRC/
                                OPIMES: .ASCIZ  / OPI/
                                NON:    .ASCIZ  / NONE/

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

```
                                                ;
                                                ; TLM UNIQUE ERROR TEXT.
                                                ;
3304  034574  052502  020123  044524  EM100:  .ASCIZ  /BUS TIME-OUT ON TLM ADDRESS/
```

```
                                                        .ASCII  /TLM STATUS WRONG/<15><12>
                                                        .ASCIZ  /CHECK THAT CABLE IS IN PLACE !!!/
                                                EM100B: .ASCII  /PROM ID WRONG/<15><12>
                                                        .ASCIZ  /CHECK THAT CORRECT PROM SET IS INSTALLED !!!/
                                                EM101:  .ASCIZ  \SYSCLK<11> AND/OR PWROK<8> NOT SET IN TLM STATUS\
                                                EM102:  .ASCIZ  /DRIVE READY STATUS BIT<0> INCORRECT IN RLCSR/
                                                EM103:  .ASCIZ  /DRIVE ERROR STATUS BITS<15:14> INCORRECT IN RLCSR/
                                                EM104:  .ASCII  /DRIVE /
                                                EM104X: .ASCIZ  /0 SELECT BITS<10:9> INCORRECT IN TLM STATUS/
                                                EM105:  .ASCIZ  /DRIVE COMMAND (GET STATUS) INCORRECT IN TLM CSR+2/
                                                EM106:  .ASCIZ  /RETURNED DRIVE STATUS INCORRECT IN RLMP/
                                                EM107:  .ASCIZ  /ERRORS IN RLCSR AFTER "GET STATUS" COMMAND/
                                                EM110:  .ASCIZ  /DRIVE COMMAND (SEEK DIFF) INCORRECT IN TLM CSR+2/
                                                EM111:  .ASCIZ  /ERRORS IN RLCSR AFTER "SEEK" COMMAND/
                                                EM112:  .ASCIZ  /WRITE GATE<13> OR DATA ACTIVE<12> DIDN'T SET DURING WRITE/
                                                EM113:  .ASCIZ  /WRITE GATE ERROR BIT<14> SET DURING WRITE/
                                                EM113A: .ASCIZ  /WRITE GATE BIT<13> STILL SET AFTER WRITE DONE/
                                                EM114:  .ASCIZ  /ERRORS IN RLCSR AFTER WRITE DATA COMMAND/
                                                EM115:  .ASCII  /ERROR STATUS INCORRECT AFTER READ DATA/
                                                EM115X: .ASCIZ  <0>/WITHOUT HEADER CHECK/
                                                EM116:  .ASCIZ  /READ HEADER CAN'T FIND TLM SECTOR 2 (CYL1,SEC6)/
                                                EM117:  .ASCII  /RECEIVED DATA INCORRECT AFTER READ/
                                                EM117X: .ASCIZ  <0>/WITHOUT HEADER CHECK/
                                                EM118:  .ASCIZ  /ERROR STATUS INCORRECT AFTER WRITE-CHECK/
                                                EM119:  .ASCII  /PROM READ FAILURE/<15><12>
                                                        .ASCII  /CAN'T FIND SECTOR /
                                                EM119X: .ASCIZ  /1/
                                                TSEC:   .ASCII  /SECTOR /
                                                TSECX:  .ASCIZ  /1 /
                                                TCLK0:  .ASCIZ  /NOMINAL CLOCK/
                                                TCLK1:  .ASCIZ  /FAST CLOCK/
                                                TCLK2:  .ASCIZ  /SLOW CLOCK/
                                                TPEAK:  .ASCIZ  / - MAX PEAK SHIFT/
                                                EXPHCRC: .ASCIZ /EXP'D: ERR HCRC OPI/
                                                EXPDCRC: .ASCIZ /EXP'D: ERR DCRC/
                                                EXPNON: .ASCIZ  /EXP'D: NONE/
                                                        .EVEN
```

```
                                        .SBTTL  DEVELOPMENT/DEBUG AIDS
                                        .SBTTL
                                        .SBTTL          RLV12 EMULATOR
                                        ;
                                        ; FOR DEBUGGING (UNDER MIMIC) WE'LL EMULATE THE ACTION
                                        ; THAT WOULD OCCUR ON A MAINT/NOP FUNCTION.
                                        ;
036704  036704                          .PRINT                  ; FOR DEBUG, MIMIC FLAG IS HERE...
036704  000000                  MIMIC:  0                       ;...SET NZ TO ENABLE EMULATOR.
                                        ;
036706  023727  003402  000000  EMURLV: CMP     RLTYP,#RL11
036714  001004                        BNE     1$              ; BR IF RLV
036716  042777  036000  144400        BIC     #36000,@RLCS    ; RL11 -- NOP -- CLEAR ERROR BITS.
036724  000517                        BR      7$              ; AND THAT'S ALL.
                                        ;
036726  027727  144400  177001  1$:   CMP     @RLMP,#-511.    ; WORD COUNT RIGHT ??
036734  001404                        BEQ     2$              ; YES.
036736  052777  112000  144360        BIS     #ERR!HNF!OPI,@RLCS ; NO SET THE ERROR BITS...
036744  000507                        BR      7$              ;...AND RETURN.
                                        ;
036746  020527  020416          2$:   CMP     R5,#BB7         ; CALLED FROM BBS7 TEST ??
036752  001004                        BNE     3$              ; SKIP IF NOT.
036754  052777  120000  144342        BIS     #ERR!NXM,@RLCS  ; YES, SET ERROR BITS...
036762  000500                        BR      7$              ;...AND RETURN.
                                        ;
036764  012701  004364          3$:   MOV     #BUF1,R1        ; STANDARD SRC...
036770  012702  005364                MOV     #BUF2,R2        ;...AND DST...
036774  012703  037236                MOV     #DFIFO,R3       ;...AND DUMMY FIFO POINTERS.
037000  023701  010352                CMP     BA16,R1         ; IS BA STANDARD (INTERNAL) ??
037004  001411                        BEQ     4$              ; BR IF SO.
037006  012701  100000                MOV     #100000,R1      ; POINT TO BUF1 IF EXT MEMORY.
037012  012702  101000                MOV     #101000,R2      ; DITTO BUF2 (RECEIVER).
037016  012737  037106  000004        MOV     #5$,ERRVEC      ; SET TRAP CATCHER.
037024  005237  177572                INC     MMR0            ;********** KT ON ***************
037030  062777  000002  144270  4$:   ADD     #2,@RLBA        ; SIMULATE INC BA...          *
037036  005577  144272                ADC     @RLBAE          ;...OVERFLOWING INTO BAE.     *
037042  012123                        MOV     (R1)+,(R3)+     ; MEM => FIFO (256 WORDS).    *
037044  000240                        240                     ;                             *
037046  020327  040236                CMP     R3,#DFIFO+1000  ;                             *
037052  103766                        BLO     4$              ;                             *
037054  012703  037236                MOV     #DFIFO,R3       ; RESET FIFO POINTER.         *
037060  062777  000002  144240  14$:  ADD     #2,@RLBA        ;                             *
037066  005577  144242                ADC     @RLBAE          ; INC BA.                     *
037072  012322                        MOV     (R3)+,(R2)+     ; FIFO => MEM (255 WORDS).    *
037074  000240                        240                     ;                             *
037076  020327  040234                CMP     R3,#DFIFO+776   ;                             *
037102  103766                        BLO     14$             ; LOOP...                     *
037104  000406                        BR      6$              ;...UNTIL DONE.               *
037106  052777  120000  144210  5$:   BIS     #ERR!NXM,@RLCS  ; ON NXM TRAP, SET ERROR BITS...*
037114  012716  037122                MOV     #6$,(SP)        ;...AND CONTINUE.             *
037120  000002                        RTI                     ;                             *
037122  005037  177572          6$:   CLR     MMR0            ;********** KT OFF **************
037126  012737  010624  000004        MOV     #TRAP4,ERRVEC   ; RESET TRAP CATCHER.
037134  017700  144170                MOV     @RLDA,R0
037140  062700  000006                ADD     #6,R0           ; ADJUST FINAL DA...
037144  110077  144160                MOVB    R0,@RLDA        ;...LO BYTE ONLY.
```

```
                                                                D 10
3399  037150  013777  003442  144154          MOV     GDCRC3,@RLMP    ; FAKE THE 1ST...
3400  037156  013737  003444  003374          MOV     GDCRC4,E.MP1    ;...AND 2ND CRC WORDS.
3401  037164  052777  000200  144132  7$:     BIS     #CRDY,@RLCS     ; SET DONE BIT...
3402  037172  032777  000100  144124          BIT     #INTEN,@RLCS    ; INTERRUPT EXPECTED ??
3403  037200  001410                          BEQ     8$              ; NO.
3404  037202                                  GETPRI  R0
  (3) 037202  104400                                                              TRAP    CSGPRI
3405  037204  042700  177437                  BIC     #^CPRI07,R0     ; MASK PRIORITY BITS.
3406  037210  020037  003340                  CMP     R0,BPRIOR
3407  037214  002002                          BGE     8$              ; BR IF CPU >= RL BR LEVEL
3408  037216  005237  010634                  INC     INTFLG          ; OTHERWISE, SET INT RECEIVED FLAG.
3409  037222  000207              8$:         RTS     PC              ;...AND RETURN.
3410
3411  037224  000005              DUMMY:      .BLKW   5               ; THESE ARE THE DUMMY REGISTERS.
3412  037236  000400              DFIFO:      .BLKW   256.            ; AND A DUMMY FIFO.
```

```
                                ;
                                ; FINALLY, A SMALL PATCH BLOCK...
                                ;
040236  040236                  PATCH:  .
        040340                          .= .+100
                                ;
                                ;...AND THAT'S ALL THERE IS TO IT !!!
                                ;
040340                                  LASTAD          ; LAST USED ADDRESS.
                                                                        .EVEN
040340  040362                                                          .WORD T$FREE
040342  000007                                                          .WORD T$SIZE
040344                          L$LAST::
                                ;*******************************************************************************
                                ;
                                ; NOW CAN A SINGLE UNIT P-TABLE, SO WE CAN RUN UNDER
                                ; DEFAULT CONDITIONS, AND CALL THE BINARY A .BIC FILE.
                                ;
040344                                  BGNSETUP 1              ; 1 UNIT ONLY...
040344                                  BGNPTAB                 ;...USING THE FOLLOWING DEFAULTS.
040344  000000                                                          .WORD   0
040346  000005                                                          .WORD   L10066-./2-1
040350                                                          L10064:
040350  174400                  .WORD   174400          ; (0) CSR ADDRESS.
040352  000160                  .WORD   160             ; (2) VECTOR.
040354  000200                  .WORD   PRI04           ; (4) PRIORITY.
040356  000000                  .WORD   0               ; (6) DRIVE (BITS 8,9,10).
040360  000003                  .WORD   3               ; (10) CONTROLLER TYPE (RLV12 W/BAE).
040362                          ENDPTAB
040362                                                          L10066:
040362                          ENDSETUP
                                ;
                                ;*******************************************************************************
000001                                  .END
```

```
ADJTN   026000              550   3076#
ADLIM   002542               98    116#
ADR   = 000020 G             147#
AFREG   030455              3175   3224#
AFTER   010074               722#   899    902
ALLREG  027100              3099   3108   3111   3114   3120   3127   3134   3151   3159   3167#
ASSEMB= 000010                17
BA16    010352               554*   771    792#  2207*  2223   2226*  2262*  2342*  2351*  3371
BA22    010354               555*   774    782    793#  2212*  2222   2227*  2261*  2343*  2352*
BB7     020416              2215#  3363
BCCFBK  003406               249#   958*   964*   967*   970    974
BCSR    003336               229#   524*   527*   556
BDDAT   003434               260#  1265*  1266*  1267   1271*  1274*  1279*  1301*  1302*  1303   1329*  1330   1352*
                            1353   1378*  1380   1408*  1409*  1410   1421*  1422*  1423   1450*  1451   1459*  1460
                            1483*  1484   1492*  1493   1520*  1521   1530*  1531   1560*  1561   1566*  1567   1574*
                            1575   1600*  1601*  1602   1607*  1608   1615*  1616   1641*  1642*  1643   1648*  1649
                            1656*  1657   1683*  1684*  1685   1690*  1691   1696*  1697   1704*  1705   1730*  1731*
                            1732   1737*  1741*  1774*  1775   1791*  1792   1836*  1840   1847   1856*  1945*  1946*
                            1951   1953   2031*  2032   2037*  2042   2047*  2048   2053*  2054   2063*  2064   2072
                            2083*  2084   2159*  2160   2168   2319*  2320   2372*  2374   2378*  2381   2386*  2389
                            2394*  2397   2407*  2413   2442*  2445   2448*  2450   2478*  2481   2511*  2514   2521*
                            2522   2524   2526   2652*  2758*  2763*  2765   3104   3107   3110   3122   3133   3157
BEFORE  010016               709#   785   2436   2472   2507   2612   2638   2729
BEGPAT  003670               278#  1322   1348   1374   1442   1478   1514
BEREG   030434              3174   3223#
BIT0  = 000001 G              58     60    100    103    110    147#   181    185   1327   2570   2695
BIT00 = 000001 GG            147#
BIT01 = 000002 GG            147#
BIT02 = 000004 GG            147#
BIT03 = 000010 G             147#
BIT04 = 000020 G             147#
BIT05 = 000040 G             147#
BIT06 = 000100 G             147#
BIT07 = 000200 G             147#
BIT08 = 000400 G             147#
BIT09 = 001000 G             147#
BIT1  = 000002 G              56    109    147#   174    176    178    180    184    350   2565   2690
BIT10 = 002000 G             147#   165   2408
BIT11 = 004000 G             147#   163    164
BIT12 = 010000 G             147#   161    162   2512
BIT13 = 020000 G             147#   159    160   2515   2523   2526
BIT14 = 040000 G             107    147#   158   2523   2575   2700
BIT15 = 100000 G             106    147#   157   3140
BIT2  = 000004 G             108    147#   175    176    179    180    186    351   2560   2685
BIT3  = 000010 G             147#   177    178    179    180    183    352
BIT4  = 000020 G             147#   189    353
BIT5  = 000040 G             147#   354
BIT6  = 000100 G             147#   171    187    188    355
BIT7  = 000200 G             147#   170
BIT8  = 000400 G             147#   167    169    356
BIT9  = 001000 G             147#   168    169    357   2408
BOE   = 000400 G             147#
BPRIOR  003340               230#   529*   568   1838   3406
BRL     002351                63     75#
BUFEND  006364               450#  2030   2983   2986   3003
BUF1    004364               448#   554    792    808    832   2059   2109   2122   2155   2226   2266   2351   2505
```

```
BUF2    005364  2646  2728  2901  2929  2943  2962  3001  3035  3368
                449#  2060  2135  2156  2602  2611  2647  2984  3005  3369
BVEC    003342   231#   528*   566
B.BA    003352   235#   711*  1779  3174
B.BAE   003360   238#   716*  3174
B.CS    003350   234#   709*   710*  1772  3174
B.DA    003354   236#   712*  1781  2036  2039  3174
B.MP    003356   237#   713*  1771*  1783  3174
CALBCC  003410   250#   926    934   939   980*
CALCRC  010714   916#  1761  1813  1842  1876  1909  1939  1984  1999  2011  2205  2281
CANTFI  025731  2959  3063#
CERR    030551   653  3227#
CHKDAT  023274  2622  2642  2646#
CKERLT  027306  3097  3102  3105  3117  3123  3154  3181#
CLKRTI  010636   571   885#
CLRTLM  024132  2431  2467  2498  2762  2782#
CONT    007050   476   552#
CPUTYP  003400   246#   504*   541  1937
CRCEND  004270   412#
CRDY  = 000200   170#   594   599   769   786   893  1251  1401  1403  1558  1590  1631  1672
                1721  2404  2419  2434  2437  2470  2473  2502  2508  2608  2613  2627  2628
                2636  2639  2725  2730  3401
CRLF    025760  3065#
CRTIM   030476   900  3225#
CSEND   004176   380#  1308  1428
CSPAT   004100   349#  1293  1399
CSRA    002323    61    73#
C$AU  = 000052    17#   628
C$AUTO= 000061    17#   588
C$BRK = 000022    17#
C$BSEG= 000004    17#  1294  1323  1349  1375  1400  1443  1479  1515  1839  2019  2133  2265
                2601  2724
C$BSUB= 000002    17#  1978  1993  2008  2106  2119
C$CEFG= 000045    17#
C$CLCK= 000062    17#
C$CLEA= 000012    17#   604
C$CLOS= 000035    17#
C$CLP1= 000006    17#  1564  1570  1605  1611  1646  1652  1688  1694  1700  1735  1740  1778
                1786  1797  1855  1891  1919  2026  2035  2045  2051  2057  2073  2147  2153
                2169  2340
C$CVEC= 000036    17#
C$DCLN= 000044    17#   513  1137  1159  1181  1203  1228  3049  3190
C$DODU= 000051    17#   587  1136  1158  1180  1202  1227  3189
C$DRPT= 000024    17#   509
C$DU  = 000053    17#   616
C$EDIT= 000003    17#    42
C$ERDF= 000055    17#   900  1232  1269  1273  1276  1281  1305  1332  1355  1382  1412  1425
                1453  1462  1486  1495  1523  1533  1563  1569  1577  1604  1610  1618  1645
                1651  1659  1687  1693  1699  1707  1734  1739  1743  1777  1785  1796  1821
                1854  1890  1918  1949  1956  2025  2034  2044  2050  2056  2070  2086  2146
                2152  2166  2224  2305  2310  2338  2376  2383  2391  2399  2415  2447  2452
                2456  2483  2487  2518  2528  2530  2534  2558  2620  2634  2644  2655  2683
                2737  2769  2771  2773
C$ERHR= 000056    17#
C$ERRO= 000060    17#
C$ERSF= 000054    17#   586  1135  1157  1179  1201  1226
```

```
C$ERSO= 000057  17#
C$ESCA= 000010  17#  1306  1333  1356  1383  1413  1426  1454  1463  1487  1496  1524  1534
C$ESEG= 000005  17#  1310  1337  1360  1387  1430  1467  1500  1538  1859  2087  2178  2341
                2662  2738
C$ESUB= 000003  17#  1989  2004  2014  2115  2128
C$ETST= 000001  17#  1138  1160  1182  1204  1233  1282  1311  1338  1361  1388  1431  1468
                1501  1539  1578  1619  1660  1708  1744  1798  1823  1861  1898  1921  1957
                2090  2182  2228  2353  2420  2457  2488  2535  2664  2740
C$EXIT= 000032  17#  1787  2015  2029  2129  2201  2253  2595  2720
C$GETB= 000026  17#
C$GETW= 000027  17#
C$GMAN= 000043  17#
C$GPHR= 000042  17#  522
C$GPLO= 000030  17#
C$GPRI= 000040  17#  3404
C$INIT= 000011  17#  573
C$INLP= 000020  17#  3181
C$MANI= 000050  17#
C$MEM = 000031  17#
C$MSG = 000023  17#  3191
C$OPEN= 000034  17#
C$PNTB= 000014  17#  2081  2177  3101  3104  3107  3110  3113  3116  3119  3122  3126  3133
                3142  3173  3174  3175  3176
C$PNTF= 000017  17#  612  624  1104  1105  1106  2845  3012  3051  3188
C$PNTS= 000016  17#
C$PNTX= 000015  17#  2072  2168  3149  3150  3157  3158  3161
C$QIO = 000377  17#
C$RDBU= 000007  17#
C$REFG= 000047  17#  465  467  469  471  473
C$RESE= 000033  17#
C$REVI= 000003  17#  42
C$RFLA= 000021  17#
C$RPT = 000025  17#  1113
C$SEFG= 000046  17#
C$SPRI= 000041  17#  603  1249  1812  1822  1840  1860  1911  1920  2140  2179
C$SVEC= 000037  17#
C$TPRI= 000013  17#
DAHS  = 000020  189#
DCRC  = 004000  163#  678  2827
DCRCME  030607  680  685  3233#
DEMES   030556  658  3228#
DERFLG  003362  239#  642*  656*
DERR  = 040000  158#  654  1252  1254  1298  1300  1405  1407  1418  1420  1591  1593  1632
                1634  1673  1675  1722  1724  2395
DFIFO   037236  3370  3381  3383  3388  3412#
DFN   = 000264  197#  900#  1232#  1269#  1273#  1276#  1281#  1305#  1332#  1355#  1382#  1412#  1425#
                1453#  1462#  1486#  1495#  1523#  1533#  1563#  1569#  1577#  1604#  1610#  1618#  1645#
                1651#  1659#  1687#  1693#  1699#  1707#  1734#  1739#  1743#  1777#  1785#  1796#  1821#
                1854#  1890#  1918#  1949#  1956#  2025#  2034#  2044#  2050#  2056#  2070#  2086#  2146#
                2152#  2166#  2224#  2305#  2310#  2338#  2376#  2383#  2391#  2399#  2415#  2447#  2452#
                2456#  2483#  2487#  2518#  2528#  2530#  2534#  2558#  2620#  2634#  2644#  2655#  2683#
                2737#  2769#  2771#  2773#
DHEAD   024570  2853*  2885#  2938*  2989
DIAGMC= 000000  17
DLT   = 010000  161#  670
DLTMES  030575  672  677  3231#
```

```
                                                                                                                                I 10
DLYCNT  003462              271#    549*    746
DMA1    016765             1678#
DMA2    017030             1993#
DMA3    017076             2008#
DMODE   024566             2884#   2945*   2948*   2954    2967
DOIT    025064             2950#
DRDY  = 000001              181#   1266    1302    1409    1422    1601    1642    1684    1731    2379    2387
DRIVE   003346              233#    530*    768
DRN     002363               65     776#
DROPPE  003464              272#    485*    511     609     611*    621     623*
DRST  = 000010              183#   2435    2444
DRTIM   030523             3226#
DS0   = 000000              166#
DS1   = 000400              167#   2416
DS2   = 001000              168#
DS3   = 001400              169#
DTABL   024572             2853    2886#
DUMMY   037224              527    3411#
EF.CON= 000036 G            147#    473
EF.NEW= 000035 G            147#    471
EF.PWR= 000034 G            147#    465
EF.RES= 000037 G            147#    469
EF.STA= 000040 G            147#    467
EMURLV  036706              789    3353#
EM0     030636              586    3238#
EM1     030706             1135    3239#
EM10    031303             2034    3249#
EM100   034574             2769    3304#
EM100A  034630             2771    3305#
EM100B  034713             2773    3307#
EM101   035007             2376    3309#
EM102   035070             2383    2391    3310#
EM103   035145             2399    3311#
EM104   035227             2415    3312#
EM104X  035235             2402*   2417*   3313#
EM105   035311             2447    3314#
EM106   035373             2452    3315#
EM107   035443             2456    3316#
EM110   035516             2483    3317#
EM111   035577             2487    3318#
EM112   035644             2518    3319#
EM113   035736             2530    3320#
EM113A  036010             2528    3321#
EM114   036066             2534    3322#
EM115   036137             2620    2644    3323#
EM115X  036205             2606*   2625*   3324#
EM116   036233             2634    3325#
EM117   036313             2655    3326#
EM117X  036355             2607*   2626*   3327#
EM118   036403             2737    3328#
EM119   036454             2558    2683    3329#
EM119X  036521             2557*   2682*   3331#
EM12    031344             2044    3250#
EM14    031405             1777    3251#
EM14A   031440             1785    3252#
EM14B   031502             1796    3253#
```

```
EM15    031544            1821    2152    3254#
EM16    031606            1854    3255#
EM17    031641            1918    3256#
EM2     030733            1157    3240#
EM20    031706            2050    3257#
EM21    031757            2056    3258#
EM22    032037            2070    2166    3259#
EM23    032103            2086    3260#
EM27    032151            1890    3261#
EM27X   032251            1879*   1880*   1895*   1896*   3263#
EM3     030760            1179    3241#
EM31    032260            1949    3264#
EM32    032306            1956    3265#
EM4     031005            1201    3242#
EM4A    031032            1226    3243#
EM4B    031060            1233    3244#
EM44    032333            1687    3266#
EM45    032366            1693    3267#
EM46    032421            1699    3268#
EM47    032454            1707    3269#
EM5     031127            1305    3245#
EM50    032510            1734    3270#
EM51    032547            1739    3271#
EM52    032573            1743    3272#
EM6     031200            1332    3246#
EM61    032617            1412    3273#
EM62    032665            1425    3274#
EM63    032735            1453    3275#
EM64    033003            1462    3276#
EM65    033053            1486    3277#
EM66    033121            1495    3278#
EM67    033171            1523    3279#
EM68    033240            1533    3280#
EM69    033311            1269    3281#
EM7     031226            1355    3247#
EM70    033345            1273    3282#
EM71    033401            1276    3283#
EM71A   033435            1281    3284#
EM72    033472            1563    3285#
EM73    033525            1569    3286#
EM73A   033560            1577    3287#
EM74    033617            1604    3288#
EM75    033651            1610    3289#
EM75A   033703            1618    3290#
EM76    033737            1645    3291#
EM77    033772            1651    3292#
EM77A   034025            1659    3293#
EM8     031254            1382    3248#
EM80    034061            2224    3294#
EM80X   034112            2202*   2204*   3295#
EM90    034147            2305    3296#
EM91    034223            2310    3297#
EM92    034300            2338    3298#
EM98    034347            2025    2146    3299#
EM99    034404             643    3113    3119    3126    3149    3300#
ENDDAT  004362             444#
```

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

```
FRMT14  030031          2072    2168    3157    3209#
FRMT15  030120          2081    2177    3210#
FRMT2   027501          3101    3201#
FRMT20  030201          3142    3211#
FRMT21  030241          3126    3212#
FRMT22  030276          3133    3213#
FRMT23  030350          3188    3214#
FRMT25  030401          3161    3216#
FRMT26  030410          3149    3217#
FRMT27  030431          3150    3158    3218#
FRMT3   027517          3113    3202#
FRMT3A  027541          3119    3203#
FRMT4   027602          3104    3110    3204#
FRMT5   027636          3107    3205#
FRMT6   027670          3116    3206#
F$AU    =  000015         17#     620     628
F$AUTO=  000020           17#     577     588
F$BGN   =  000040           17#      55      97      99     456     464     577     592     608     620    1099    1107    1126
                         1138    1148    1160    1170    1182    1192    1204    1215    1233    1248    1282    1292    1294
                         1306    1311    1321    1323    1333    1338    1347    1349    1356    1361    1371    1375    1383
                         1388    1398    1400    1413    1426    1431    1441    1443    1454    1463    1468    1477    1479
                         1487    1496    1501    1511    1515    1524    1534    1539    1552    1578    1589    1619    1630
                         1660    1671    1708    1718    1744    1755    1787    1798    1810    1823    1835    1839    1861
                         1871    1898    1908    1921    1936    1957    1974    1978    1989    1993    2004    2008    2014
                         2015    2019    2029    2090    2102    2106    2115    2119    2128    2129    2133    2182    2196
                         2201    2228    2247    2253    2265    2353    2367    2420    2428    2457    2464    2488    2495
                         2535    2546    2595    2601    2664    2671    2720    2724    2740    3096    3429    3430    3436
                         3437
F$CLEA=  000007           17#     592     604
F$DU    =  000016         17#     608     616
F$END   =  000041           17#      66     112     573     588     604     616     628    1107    1113    1126    1138    1148
                         1160    1170    1182    1192    1204    1212    1233    1248    1282    1292    1306    1310    1311
                         1321    1333    1337    1338    1347    1356    1360    1361    1371    1383    1387    1388    1398
                         1413    1426    1430    1431    1441    1454    1463    1467    1468    1477    1487    1496    1500
                         1501    1511    1524    1534    1538    1539    1552    1578    1589    1619    1630    1660    1671
                         1708    1718    1744    1755    1787    1798    1810    1823    1835    1859    1861    1871    1898
                         1908    1921    1936    1957    1974    1978    1989    1993    2004    2008    2014    2015    2029
                         2087    2090    2102    2106    2115    2119    2128    2129    2178    2182    2196    2201    2228
                         2247    2253    2341    2353    2367    2420    2428    2457    2464    2488    2495    2535    2546
                         2595    2662    2664    2671    2720    2738    2740    3191    3429    3430    3436    3437
F$HARD=  000004           17#      55      57      59      64      66     101     104
F$HW    =  000013         17#      81      91
F$INIT=  000006           17#     464     573
F$JMP   =  000050         17#    1107    1787    2015    2029    2129    2201    2253    2595    2720
F$MOD   =  000000         17#
F$MSG   =  000011         17#    3096    3191
F$PROT=  000021           17#     456     460
F$PWR   =  000017         17#
F$RPT   =  000012         17#    1099    1113
F$SEG   =  000003         17#    1294    1310    1323    1337    1349    1360    1375    1387    1400    1430    1443    1467
                         1479    1500    1515    1538    1839    1859    2019    2087    2133    2178    2265    2341    2601
                         2662    2724    2738
F$SOFT=  000005           17#      97     101     104     112
F$SRV   =  000010         17#
F$SUB   =  000002         17#    1978    1989    1993    2004    2008    2014    2106    2115    2119    2128
F$SW    =  000014         17#     135     143
```

```
FSTEST= 000001          17#  1126  1138  1148  1160  1170  1182  1192  1204  1212  1233  1248  1282
                      1292   1311  1321  1338  1347  1361  1371  1388  1398  1431  1441  1468  1477
                      1501   1511  1539  1552  1578  1589  1619  1630  1660  1671  1708  1718  1744
                      1755   1798  1810  1823  1835  1861  1871  1898  1908  1921  1936  1957  1974
                      2090   2102  2182  2196  2228  2247  2353  2367  2420  2428  2457  2464  2488
                      2495   2535  2546  2664  2671  2740

GDCRC3  003442        263#   926*  2046  3399
GDCRC4  003444        264#   939*  2052  3400
GDDAT   003432        259#   585*  1134* 1156* 1178* 1200* 1216* 1251* 1254* 1267  1270* 1295* 1296*
                      1297  1300*  1303  1324* 1327* 1328  1330  1350* 1351  1353  1376* 1377  1379*
                      1380  1402*  1403* 1407* 1410  1415* 1416* 1420* 1423  1445* 1448* 1451  1456*
                      1457* 1460   1481* 1484  1489* 1490* 1493  1517* 1518* 1521  1526* 1527* 1528*
                      1531  1559*  1561  1565* 1567  1573* 1575  1590* 1593* 1594  1602  1606* 1608
                      1614* 1616   1631* 1634* 1635  1643  1647* 1649  1655* 1657  1672* 1675* 1676
                      1685  1689*  1691  1695* 1697  1703* 1705  1721* 1724* 1725  1729* 1732  1736*
                      1772* 1773*  1775  1838* 1847  2030* 2032  2036* 2041* 2042  2046* 2048  2052*
                      2054  2062*  2064  2072  2082* 2084  2158* 2160  2168  2318* 2320  2373* 2374
                      2380* 2381   2388* 2389  2396* 2397  2412* 2413  2444* 2445  2449* 2450  2480*
                      2481  2513*  2514  2522* 2523* 2524  2651* 2750* 2757* 2764* 2765  3101  3104
                      3110  3133   3157

GETERR  007510        641#  1888  2023  2144  2220  2303  2308  2453  2484  2531  2617  2641  2734
GSBIT = 000002        184#  2435  2444
GSTAT = 000004        175#  2434
G$CNTO= 000200          17#
G$DELM= 000372          17#
G$DISP= 000003          17#   99
G$EXCP= 000400          17#
G$HILI= 000002          17#
G$LOLI= 000001          17#
G$NO  = 000000          17#
G$OFFS= 000400          17#    56    58    60    61    62    63    65    98   100   102   103   105
                      106   107   108   109   110   111
G$OFSI= 000376          17#    56    58    60    61    62    63    65    98   100   102   103   105
                      106   107   108   109   110   111
G$PRMA= 000001          17#    61    62   102
G$PRMD= 000002          17#    63    65    98   105   111
G$PRML= 000000          17#    56    58    60   100   103   106   107   108   109   110
G$RADA= 000140          17#
G$RADB= 000000          17#
G$RADD= 000040          17#    98
G$RADL= 000120          17#    56    58    60   100   103   106   107   108   109   110
G$RADO= 000020          17#    61    62    63    65   102   105   111
G$XFER= 000004          17#    57    59    64   101   104
G$YES = 000010          17#    56    58    60    61    62    63    65    98   100   102   103   105
                      106   107   108   109   110   111

HCRC  = 004000        164#   678  2833
HCRCME  030615         683  3234#
HEADR   025670        2952  3060#
HEXMOD  025350        2948  3015#
HEXTXT  025763        3016  3031  3066#
HINUM   003452         267#  483*  516   553*  835   858   866   871*
HNF   = 010000         162#  670  1885  3360
HNFMES  030602         675  3232#
HOE   = 100000 G       147#
HSECT   025714        2935* 3062#
HUNIT   025701        2897  2915*  2921  3061#
```

```
IBE    = 010000 G  147#
IDU    = 000040 G  147#
IER    = 020000 G  147#
INHFC    003177    109   128#
INHMNP   003120    107   126#
INHMXP   003070    106   125#
INHNC    003150    108   127#
INHSC    003223    110   129#
INIDA    003440    262#  783   916*  917   2610  2727  2822*  2826*  2832*  2838*  2842*  2850*
INIMP    003436    261#  713   766*  784   2432* 2468* 2503*  2504   2549*  2609   2637   2674*  2726
INTEN  = 000100    171#  1816  1845  1914  2145  3402
INTFLG   010634    884#  1811* 1818  1841* 1849  1852  1912*  1916   2139*  2150   3408*
IOPAG    021252   2254*  2260* 2346#
ISR    = 000100 G  147#
IXE    = 004000 G  147#
I$AU   = 000041     17#   620#  628#
I$AUTO= 000041     17#   577#  588#
I$CLN  = 000041     17#   592#  604#
I$DU   = 000041     17#   608#  616#
I$HRD  = 000041     55#    66#   99
I$INIT= 000041     17#   464#  573#
I$MOD  = 000041     17#
I$MSG  = 000041     17#  3096#  3191#
I$PROT= 000040     17#   456#
I$PTAB= 000041     17#  3430#  3436#
I$PWR  = 000041     17#
I$RPT  = 000041     17#  1099#  1113#
I$SEG  = 000041     17#  1126   1148  1170  1192  1212  1248  1292  1294#  1306   1310#  1321   1323#
                  1333   1337#  1347  1349#  1356  1360#  1371  1375#  1383   1387#  1398   1400#  1413
                  1426   1430#  1441  1443#  1454  1463  1467#  1477  1479#  1487   1496   1500#  1511
                  1515#  1524   1534  1538#  1552  1589  1630  1671  1718   1755   1810   1835   1839#
                  1859#  1871   1908  1936  1974  1978  1993  2008  2019#  2029   2087#  2102   2106
                  2119   2133#  2178# 2196  2247  2265# 2341# 2367  2428   2464   2495   2546   2601#
                  2662#  2671   2724# 2738#
I$SETU= 000041     17#  3429#  3430  3437#
I$SFT  = 000041     97#    99   112#
I$SRV  = 000041     17#
I$SUB  = 000041     17#  1126   1148  1170  1192  1212  1248  1292  1321   1347   1371   1398   1441
                  1477   1511   1552  1589  1630  1671  1718  1755  1810   1835   1871   1908   1936
                  1974   1978#  1989# 1993# 2004# 2008# 2014# 2102  2106#  2115#  2119#  2128#  2196
                  2247   2367   2428  2464  2495  2546  2671
I$TST  = 000041     17#  1126#  1138# 1148# 1160# 1170# 1182# 1192#  1204#  1212#  1233#  1248#  1282#
                  1292#  1311#  1321# 1338# 1347# 1361# 1371# 1388#  1398#  1431#  1441#  1468#  1477#
                  1501#  1511#  1539# 1552# 1578# 1589# 1619# 1630#  1660#  1671#  1708#  1718#  1744#
                  1755#  1787   1798# 1810# 1823# 1835# 1861# 1871#  1898#  1908#  1921#  1936#  1957#
                  1974#  1978   1993  2008  2015  2090# 2102# 2106   2119   2129   2182#  2196#  2201
                  2228#  2247#  2253  2353# 2367# 2420# 2428# 2457#  2464#  2488#  2495#  2535#  2546#
                  2595   2664#  2671# 2720  2740#
JSJMP  = 000167     17#  1107
KBDIN    025432   2900   2928  2942  3035#
KBDOUT   025476   2898   2922  2926  2940  2951  2958  2965  3030  3051#
KIPARO= 172340   1010#  1011  1042  2246
KIPDRO= 172300   1011#  1043
LDFUN    010210    765#  1763  1815  1844  1881  1913  1942  2020  2141   2213   2283
LOE    = 040000 G  147#
LONUM    003454    268#  484*  515   552*  857   864   870*
```

```
LOT   = 000010 G          147#
LSI     006574            494     504#
L$ACP   002110 G           42#
L$APT   002036 G           42#
L$AU    007432 G           42     620#
L$AUT   002070 G           42#
L$AUTO  007214 G           42     577#
L$CCP   002106 G           42#
L$CLEA  007272 G           42     592#
L$CO    002032 G           42#
L$DEPO  002011 G           42#
L$DESC  002122 G           42      44#
L$DESP  002076 G           42#
L$DEVP  002060 G           42#
L$DISP  026052 G           42    3087*   3090#
L$DLY   002116 G           42#
L$DTP   002040 G           42#
L$DTYP  002034 G           42#
L$DU    007350 G           42     608#
L$DUT   002072 G           42#
L$DVTY  002142 G           42      46#
L$EF    002052 G           42#
L$ENVI  002044 G           42#
L$ETP   002102 G           42#
L$EXP1  002046 G           42#
L$EXP4  002064 G           42#
L$EXP5  002066 G           42#
L$HARD  002172 G           42      55#
L$HINE  002120 G           42#
L$HPCP  002016 G           42#
L$HPTP  002022 G           42#
L$HW    002376 G           42      81#
L$ICP   002104 G           42#
L$INIT  006374 G           42     464#
L$LADP  002026 G           42#
L$LAST  040344 G           37      42     3422#   3437
L$LOAD  002100 G           42#
L$LUN   002074 G           42#
L$MREV  002050 G           42#
L$NAME  002000 G           42#
L$PRIO  002042 G           42#
L$PROT  006366 G           42     456#
L$PRT   002112 G           42#
L$REPP  002062 G           42#
L$REV   002010 G           42#
L$RPT   011644 G           42    1099#
L$SOFT  002412 G           42      97#
L$SPC   002056 G           42#
L$SPCP  002020 G           42#
L$SPTP  002024 G           42#
L$STA   002030 G           42#
L$SW    003304 G           42     135#
L$TEST  002114 G           42#
L$TIML  002014 G           42#
L$UNIT  002012 G           42#    511     520     578     609
L10000  002266             55      66#
```

```
L10001  002410              81     91#
L10002  002542              97    112#
L10003  003322             135    143#
L10005  007212             573#
L10006  007270             588#
L10007  007346             604#
L10010  007430             616#
L10011  007506             628#
L10012  012112            1107   1113#
L10013  012176            1138#
L10014  012262            1160#
L10015  012346            1182#
L10016  012432            1204#
L10017  012554            1233#
L10020  013016            1282#
L10021  013136            1311#
L10022  013242            1336#
L10023  013330            1361#
L10024  013434            1388#
L10025  013662            1431#
L10026  014046            1468#
L10027  014214            1501#
L10030  014406            1539#
L10031  014610            1578#
L10032  015034            1619#
L10033  015260            1660#
L10034  015550            1708#
L10035  015744            1744#
L10036  016156            1787   1798#
L10037  016244            1823#
L10040  016412            1861#
L10041  016546            1898#
L10042  016632            1921#
L10043  016760            1957#
L10044  017660            2015   2090#
L10045  017026            1989#
L10046  017074            2004#
L10047  017124            2014#
L10050  020304            2129   2182#
L10051  017712            2115#
L10052  017746            2128#
L10053  020500            2201   2228#
L10054  021274            2253   2353#
L10055  021646            2420#
L10056  022052            2457#
L10057  022220            2488#
L10060  022500            2535#
L10061  023362            2595   2664#
L10062  023760            2720   2740#
L10063  027364            3191#
L10064  040350            3430#
L10066  040362            3430   3436#
MAINT =  000000             173#  1882   1914   1943   2021   2142   2214   2284
MFPT  =  000007             499#   500
MIMIC    036704             525    726    787   3351#
MK    =  000001             185#  2435   2444   2471   2480
```

```
MMR0   = 177572          1006#  1026   1058*  1079*  2249   2271*  2279*  2317*  2324*  3376*  3394*
MMR1   = 177574          1007#
MMR2   = 177576          1008#
MMR3   = 172516          1009#  1055   1056*  1063*  2259*
MPXCLK   003312           139#  2560   2565   2570   2575   2579   2685   2690   2695   2700   2704
MSGNOT   012104          1100*  1103*  1104   1111#
MSG2     011764          1104   1108#
MSG3     012014          1105   1109#
MSG4     012047          1106   1110#
NEWPAS   006620           472    511#   521
NLSI     006570           498    503#
NOANS    025530          2921*  2923   3055#
NON      030630           647   3236#
NOOP   = 000000           172#
NOSIG    026160          2558   2683   3097#
NXM    = 020000           159#    662   2027   2148   2216   2300   3365   3391
NXMMES   030563           664    669   3229#
NXTU     006662           475    518#   523
OCTMOD   025324          2884   2945   3012#
OCTTXT   025770          3012   3067#
OPI    = 002000           165#    665    673    681    686   1794   1885   1947   2027   2148   2833   3360
OPIMES   030623           689   3235#
OPIMN    003446           265#   1953
OPIMX    003450           266#   1951
O$APTS= 000000            17#     42
O$AU   = 000001           17#     41#    42
O$BGNR= 000001            17#     41#    42
O$BGNS= 000001            17#     41#    42
O$DU   = 000001           17#     41#    42
O$ERRT= 000000            17#     42
O$GNSW= 000001            17#     41#    42
O$POIN= 000001            17#     41#    42
O$SETU= 000001            17#     41#    42   3422
PAR    = 020000          160#    662
PARMES   030570           667   3230#
PAR4   = 172350          2246#  2263*  2329   2344*  2345
PATCH    040236          3417#
PATCRC   004200           384#   1979   1994
PATDAT   004272           416#
PDSW     003306           137#    480   2893*
PNT    = 001000 G         147#
PRGSIZ= 000100            37#
PRI    = 002000 G         147#
PRI00  = 000000 G         147#   1812   1911   2140
PRI01  = 000040 G         147#
PRI02  = 000100 G         147#
PRI03  = 000140 G         147#
PRI04  = 000200 G          84    147#   3433
PRI05  = 000240 G         147#
PRI06  = 000300 G         147#
PRI07  = 000340 G         147#    570    572    603   1249   1822   1836   1860   1920   2179   3405
PRM      003047           105    124#
PRMDMP   003024           103    123#
PROMID   003310           138#   2764
PRTPA    026610          3125   3129   3132   3138#
PSERR    024506          2615   2732   2827*  2833*  2851*  2859#
```

```
                                                                E 11
PSETXT  024504          2619    2621    2736    2828*   2834*   2852*   2858#
PSX     024214          2554    2679    2806#
PS1     024252          2811    2821#
PS2     024270          2812    2825#
PS3     024322          2813    2831#
PS4     024354          2814    2837#
PS5     024372          2815    2841#
PS6     024442          2816    2849#
PS7     024410          2817    2845#
PURE    025774          3051    3068#
Q0      025561          2899    3056#
Q1      025612          2927    3058#
Q2      025637          2941    3059#
RAND    010526           834     854#
RDHDR = 000010           177#    2627
RDNHDR= 000016           180#    2636
READ  = 000014           179#    2608
READDU  023356          2635    2662#
READNH  023130          2624#
READSE  022766          2578    2583    2601#
RESTAR  006452           466     470     478#
RHHS  = 000100           187#
RLBA    003326           225#     559*    711     723     771*   1150*   1152    1156    1255*   1271    1328*   1329    1444*
                        1449*   1450    1455*   1458*   1459    1553*   1560    1599*   1636*   1648    1677*   1690    1726*
                        1737    2505*   2611*   2728*   2789*   3377*   3384*
RLBAE   003334           228#     565*    602*    716     731     782*   1216    1217*   1219    1259*   1279    1377*   1378
                        1516*   1519*   1520    1525*   1529*   1530    1557*   1574    1598*   1615    1639*   1656    1681*
                        1704    1728*   2793*   3378*   3385*
RLCS    003324           224#     557*    581     585     594     599*    709     722     770*    780*    786*    893    1128*
                        1130    1134    1250*   1252    1265    1297*   1298    1301    1401*   1404*   1405    1408    1414*
                        1417*   1418    1421    1558*   1591    1594*   1600    1632    1635*   1641    1673    1676*   1683
                        1722    1725*   1730    1947    2377    2385    2393    2404*   2405*   2419*   2434*   2437*   2470*
                        2473*   2502*   2508*   2608*   2613*   2627*   2628*   2629    2636*   2639*   2725*   2730*   3355*
                        3360*   3365*   3391*   3401*   3402
RLDA    003330           226#     561*    712     724     783*   1172*   1174    1178    1256*   1274    1351*   1352    1480*
                        1482*   1483    1488*   1491*   1492    1554*   1566    1595*   1607    1640*   1678*   1696    1727*
                        1741    2435*   2471*   2506*   2610*   2727*   3396    3398*
RLINT   010632           567     883#
RLMP    003352           227#     563*    725     728     784*   1194*   1196    1200    1682*   2504*   2609*   2631    2637*
                        2726*   2790*   3358    3399*
RLTYP   003402           247#     531*    600     714     729     772    1213    1221    1257    1277    1325    1372    1446
                        1512    1555    1571    1596    1612    1637    1653    1679    1701    1719    1766    2197    2257
                        2286    2791    3076    3168    3353
RLV11 = 000001           153#    2197
RLV12 = 000002           154#    1213    3078
RLV12X= 000003           155#     600     714     729     772    1221    1257    1277    1372    1512    1555    1571    1596
                        1612    1637    1653    1679    1701    1719    2257    2286    2791    3168
RL1     002266            56      70#
RL1A    002276            58      71#
RL11  = 000000           152#    1325    1446    1766    3353
RL2     002313            60      72#
ROMDUM  024606           482    2893#    2924    2960    2974
SEC1    003247           111     130#
SEEK  = 000006           176#    2470
SETCMP  010362           804#    1996
SETPAT  010356           802#    1981
```

```
SETRAN  010452          830#   2009   2264
SFN     = 000006        196#    586#  1135#  1157#  1179#  1201#  1226#
SIGN    = 000004        186#
SIMBCC  011060          922     930    935    952#
SKP1    = 000401        217#   2748
SKP2    = 000402        218#    502    547    901   2203   2294   2519   3083   3085
SKP3    = 000403        219#
SNGLSE  003314          140#   2550   2591   2675   2716   2846*
START   006452          468     478#
STHS    = 000100        188#
SVCGBL= 000000           17#     19#    38#    42     44     46     48#    55     81     97    135    456    464
                        577     592    608    620   1099   3090   3096   3422#
SVCINS= 000001           17#     23#    39#    42     44     46     49#    55     56     57     58     59     60
                         61      62     63     64     65     66     81     97     98     99    100    101    102
                        103     104    105    106    107    108    109    110    111    112    135    465    466
                        467     468    469    470    471    472    473    474    509    513    522    523    573
                        586     587    588    603    604    612    616    624    628    900   1104   1105   1106
                       1107    1113   1135   1136   1137   1138   1157   1158   1159   1160   1179   1180   1181
                       1182    1201   1202   1203   1204   1226   1227   1228   1232   1233   1249   1269   1273
                       1276    1281   1282   1294   1305   1306   1310   1311   1323   1332   1333   1337   1338
                       1349    1355   1356   1360   1361   1375   1382   1383   1387   1388   1400   1412   1413
                       1425    1426   1430   1431   1443   1453   1454   1462   1463   1467   1468   1478   1486
                       1487    1495   1496   1500   1501   1515   1523   1524   1533   1534   1538   1539   1563
                       1564    1569   1570   1577   1578   1604   1605   1610   1611   1618   1619   1645   1646
                       1651    1652   1659   1660   1687   1688   1693   1694   1699   1700   1707   1708   1734
                       1735    1739   1740   1743   1744   1777   1778   1785   1786   1787   1796   1797   1798
                       1812    1821   1822   1823   1839   1840   1854   1855   1859   1860   1861   1890   1891
                       1898    1911   1918   1919   1920   1921   1949   1956   1957   1978   1989   1993   2004
                       2008    2014   2015   2019   2025   2026   2029   2034   2035   2044   2045   2050   2051
                       2056    2057   2070   2072   2073   2081   2086   2087   2090   2106   2115   2119   2128
                       2129    2133   2140   2146   2147   2152   2153   2166   2168   2169   2177   2178   2179
                       2182    2201   2224   2228   2253   2262   2305   2310   2338   2340   2341   2353   2376
                       2385    2391   2399   2415   2420   2447   2452   2456   2457   2483   2487   2488   2518
                       2528    2530   2534   2535   2558   2595   2601   2620   2634   2644   2655   2662   2664
                       2683    2720   2724   2737   2738   2740   2769   2771   2773   2845   3012   3049   3051
                       3090    3101   3104   3107   3110   3113   3116   3119   3122   3126   3133   3142   3149
                       3150    3157   3158   3161   3173   3174   3175   3176   3181   3182   3188   3189   3190
                       3191    3404   3422   3430
SVCSUB= 000001           17#     21#  1978   1993   2008   2106   2119
SVCTAG= 000001           17#     22#    66     91    112    143    573    588    604    616    628   1113   1120
                       1125    1138   1142   1147   1160   1164   1169   1182   1186   1191   1204   1208   1211
                       1235    1237   1247   1282   1286   1291   1310   1311   1315   1320   1337   1338   1342
                       1376    1360   1361   1365   1370   1387   1388   1392   1397   1430   1431   1435   1440
                       1467    1468   1472   1476   1500   1501   1505   1510   1538   1539   1543   1551   1578
                       1582    1588   1619   1623   1629   1660   1664   1670   1708   1712   1717   1744   1748
                       1754    1798   1802   1809   1823   1827   1854   1859   1861   1865   1870   1898   1903
                       1907    1921   1925   1935   1957   1961   1973   1989   2004   2014   2087   2090   2094
                       2101    2115   2128   2178   2182   2186   2195   2228   2232   2245   2341   2353   2358
                       2366    2420   2424   2427   2437   2461   2463   2488   2492   2494   2535   2539   2545
                       2662    2664   2668   2670   2738   2740   3191   3430   3436
SVCTST= 000001           17#     20#  1120   1148   1170   1192   1212   1248   1292   1321   1347   1371   1398
                       1441    1477   1511   1552   1589   1630   1671   1718   1755   1810   1835   1871   1908
                       1936    1974   2102   2196   2247   2367   2428   2464   2495   2546   2671
S$LSYM= 010000           17#     66#    91#   112#   143#   573#   588#   604#   616#   628#  1113#  1138#  1160#
                       1182#   1204#  1233#  1282#  1294#  1311#  1323#  1338#  1349#  1361#  1375#  1388#  1400#
                       1431#   1443#  1468#  1479#  1501#  1515#  1539#  1578#  1619#  1660#  1708#  1744#  1798#
```

```
                    1823#  1839#  1861#  1898#  1921#  1957#  1989#  2004#  2014#  2019#  2090#  2115#  2128#
                    2133#  2182#  2228#  2265#  2353#  2420#  2457#  2488#  2535#  2601#  2664#  2724#  2740#
                    3191#
TAB       025756     2966   3064#
TCLK0     036536     2683   2688   3334#
TCLK1     036554     2568   2693   3335#
TCLK2     036567     2573   2698   3336#
TCSR      003344      232#   535*   535*   538*  2370*  2371*  2372   2384*  2392*  2400*  2406   2750   2753*
                    2758   2782   3080
TCSUO     002776      102    122#
TCSO      003320      142#   535   2909
TEMHI     003460      270#   516*   553
TEMLO     003456      269#   515*   552   2010
TEMP1     003422      255#   917*   918    920*   921    928*   929
TEMP2     003412      251#   955*   978*
TEMP3     003414      252#   956*   960*
TEMP4     003416      253#   957*   959    969*   971    976*   977*   980
TEMP5     003420      254#   918*   919*   920    927*   928
TLMBP     002575       99    118#
TLMF      003316      141#   535
TLMIN     002751      100    121#
TLMOK     023762     2368   2429   2465   2496   2747#
TLMPID    023766     2547   2672   2749#
TLM1      021276     2367#
TLM2      021650     2428#
TLM3      022054     2464#
TLM4      022222     2495#
TLM5      022502     2546#
TLM6      023364     2671#
TMP0      003424      256#   807*   810    813*  1820*  1837*  1857*  2038*  2039*  2040*  2041   2066*  2072
                    2162*  2168   2327*  2653*  2950*  2953   3116   3133   3157
TMP1      003426      257#  2067*  2072   2163*  2168   2222*  2296*  2330*  2332*  2337*  2455*  2486*  2533*
                    2619*  2643*  2654*  2736*  3141*  3142   3149   3157
TMP2      003430      258#  2058*  2068   2071*  2079   2081   2154*  2164   2167*  2175   2177   2223*  2297*
                    2329*  2331*  2336*  2563*  2568*  2573*  2688*  2693*  2698*  3138   3140*  3142   3161
TPEAK     036602     2577*  2582*  2702*  2707*  3161   3337#
TRAP4     010624      569    879#  2280   3395
TRPFLG    010626      580*   583    880#  1127*  1132   1149*  1154   1171*  1176   1193*  1198   1215*  1224
                    1230   2248*  2251   2270*  2276*  2298   2752*  2755
TSEC      036523     3161   3332#
TSECX     036532     2557   2682   2821*  2825*  2831*  2837*  2841*  2849*  3333#
TURKY     024510     2845   2861#
TSARGC=   000001       42#   612#   624#  1104#  1105#  1106#  2072#  2081#  2168#  2177#  2845#  3012#  3051#
                    3101#  3104#  3107#  3110#  3113#  3116#  3119#  3122#  3126#  3133#  3142#  3149#  3150#
                    3157#  3158#  3161#  3173#  3174#  3175#  3176#  3188#
TSCODE=   004032       56#    57#    58#    59#    60#    61#    62#    63#    64#    65#    98#    99#   100#
                     101#   102#   103#   104#   105#   106#   107#   108#   109#   110#   111#
TSERRN=   000264        17#   506#   900#  1135#  1157#  1179#  1201#  1226#  1232#  1269#  1273#  1276#  1281#
                    1305#  1332#  1355#  1382#  1412#  1425#  1453#  1462#  1486#  1495#  1523#  1533#  1563#
                    1569#  1577#  1604#  1610#  1618#  1645#  1651#  1659#  1687#  1693#  1699#  1707#  1734#
                    1739#  1743#  1777#  1785#  1796#  1821#  1854#  1890#  1918#  1949#  1956#  2025#  2034#
                    2044#  2050#  2056#  2070#  2086#  2146#  2152#  2166#  2224#  2305#  2310#  2338#  2376#
                    2383#  2391#  2399#  2415#  2447#  2452#  2456#  2483#  2487#  2518#  2528#  2530#  2534#
                    2558#  2620#  2634#  2644#  2655#  2683#  2757#  2769#  2771#  2773#
TSEXCP=   000000       61#    62#    63#    65#    98#   102#   105#   111#
TSFLAG=   000040     1107#  1306#  1333#  1356#  1383#  1413#  1426#  1454#  1463#  1487#  1496#  1524#  1534#
```

```
                   1787#   2015#   2029#   2129#   2201#   2253#   2595#   2720#
T$FREE= 040362     3422    3437#
T$GMAN= 000000      17#
T$HILI= 000006      61#     62#     63#     65#     98#    102#    105#    111#
T$LAST= 000001      17#   3422#   3429
T$LOLI= 000000      61#     62#     63#     65#     98#    102#    105#    111#
T$LSYM= 010000      17#     66      91     112     143     573     588     604     616     628    1113    1138    1160
                   1182    1204    1233    1282    1311    1338    1361    1388    1431    1468    1501    1539    1578
                   1619    1660    1708    1744    1798    1823    1861    1898    1921    1957    1989    2004    2014
                   2090    2115    2128    2182    2228    2353    2420    2457    2488    2535    2664    2740    3191
T$LTNO= 000043     3422#
T$NEST= 177777      17#     55#     57      59      64      66#     81#     91#     97#    101     104     112#    135#
                    143#    456#    460#    464#    573#    577#    588#    592#    604#    608#    616#    620#    628#
                   1099#   1113#   1126#   1138#   1148#   1160#   1170#   1182#   1192#   1204#   1212#   1233#   1248#
                   1282#   1292#   1294#   1310#   1311#   1321#   1323#   1337#   1338#   1347#   1349#   1360#   1361#
                   1371#   1375#   1387#   1388#   1398#   1400#   1430#   1431#   1441#   1443#   1467#   1468#   1477#
                   1479#   1500#   1501#   1511#   1515#   1538#   1539#   1552#   1578#   1589#   1619#   1630#   1660#
                   1671#   1708#   1718#   1744#   1755#   1798#   1810#   1823#   1835#   1839#   1859#   1861#   1871#
                   1898#   1908#   1921#   1936#   1957#   1974#   1978#   1989#   1993#   2004#   2008#   2014#   2019#
                   2087#   2090#   2102#   2106#   2115#   2119#   2128#   2133#   2178#   2182#   2196#   2228#   2247#
                   2265#   2341#   2353#   2367#   2420#   2428#   2457#   2464#   2488#   2495#   2535#   2546#   2601#
                   2662#   2664#   2671#   2724#   2738#   2740#   3096#   3191#
T$NSO = 000011      55#     57      59      64      66      81#     91      97#    101     104     112     135#    143
                    456#    460     464#    573     577#    588     592#    604     608#    616     620#    628    1099#
                   1113    1126#   1138    1148#   1160    1170#   1182    1192#   1204    1212#   1233    1248#   1282
                   1292#   1311    1321#   1338    1347#   1361    1371#   1388    1398#   1431    1441#   1468    1477#
                   1501    1511#   1539    1552#   1578    1589#   1619    1630#   1660    1671#   1708    1718#   1744
                   1755#   1798    1810#   1823    1835#   1861    1871#   1898    1908#   1921    1936#   1957    1974#
                   2090    2102#   2182    2196#   2228    2247#   2353    2367#   2420    2428#   2457    2464#   2488
                   2495#   2535    2546#   2664    2671#   2740    3096#   3191
T$NS1 = 000003     1294#   1310    1323#   1337    1349#   1360    1375#   1387    1400#   1430    1443#   1467    1479#
                   1500    1515#   1538    1839#   1859    1978#   1989    1993#   2004    2008#   2014    2019#   2087
                   2106#   2115    2119#   2128    2133#   2178    2265#   2341    2601#   2662    2724#   2738
T$PCNT= 000000     3429#   3430#
T$PTAB= 010065     3430#
T$PTHV= 000001       42    3437#
T$PTNU= 000001      17#   3430#   3437
T$SAVL= 177777      17#
T$SEGL= 177777      17#   1294#   1306    1310#   1323#   1333    1337#   1349#   1356    1360#   1375#   1383    1387#
                   1400#   1413    1426    1430#   1443#   1454    1463    1467#   1479#   1487    1496    1500#   1515#
                   1524    1534    1538#   1839#   1859#   2019#   2029    2087#   2133#   2178#   2265#   2341#   2601#
                   2662#   2724#   2738#
T$SEKO= 010000     1294#   1306    1310    1323#   1333    1337    1349#   1356    1360    1375#   1383    1387    1400#
                   1413    1426    1430    1443#   1454    1463    1467    1479#   1487    1496    1500    1515#   1524
                   1534    1538    1839#   1859    2019#   2029    2087    2133#   2178    2265#   2341    2601#   2662
                   2724#   2738
T$SIZE= 000007     3422    3437#
T$SUBN= 000000      17#   1126#   1148#   1170#   1192#   1212#   1248#   1292#   1321#   1347#   1371#   1398#   1441#
                   1477#   1511#   1552#   1589#   1630#   1671#   1718#   1755#   1810#   1835#   1871#   1908#   1936#
                   1974#   1978#   1993#   2008#   2102#   2106#   2119#   2196#   2247#   2367#   2428#   2464#   2495#
                   2546#   2671#
T$TAGL= 177777      17#
T$TAGN= 010067      17#     55#     81#     97#    135#    456#    464#    577#    592#    608#    620#   1099#   1126#
                   1148#   1170#   1192#   1212#   1248#   1292#   1321#   1347#   1371#   1398#   1441#   1477#   1511#
                   1552#   1589#   1630#   1671#   1718#   1755#   1810#   1835#   1871#   1908#   1936#   1974#   1978#
                   1993#   2008#   2102#   2106#   2119#   2196#   2247#   2367#   2428#   2464#   2495#   2546#   2671#
```

```
                  3096#  3429#  3430#
TSTEMP= 000011      56#    58#    60#    61#    62#    63#    65#    66#    91#    98#   100#   102#   103#
                   105#   106#   107#   108#   109#   110#   111#   112#   143#   460#   573#   588#   604#
                   616#   628#  1107#  1113#  1120#  1125#  1138#  1142#  1147#  1160#  1164#  1169#  1182#
                  1186#  1191#  1204#  1208#  1211#  1233#  1237#  1247#  1282#  1286#  1291#  1306#  1310#
                  1311#  1315#  1320#  1333#  1337#  1338#  1342#  1346#  1356#  1360#  1361#  1365#  1370#
                  1383#  1387#  1388#  1392#  1397#  1413#  1426#  1430#  1431#  1435#  1440#  1454#  1463#
                  1467#  1468#  1472#  1476#  1487#  1496#  1500#  1501#  1505#  1510#  1524#  1534#  1538#
                  1539#  1543#  1551#  1578#  1582#  1588#  1619#  1623#  1629#  1660#  1664#  1670#  1708#
                  1712#  1717#  1744#  1748#  1754#  1787#  1798#  1802#  1809#  1823#  1827#  1834#  1859#
                  1861#  1865#  1870#  1898#  1903#  1907#  1921#  1925#  1935#  1957#  1961#  1973#  1989#
                  2004#  2014#  2015#  2029#  2087#  2090#  2094#  2101#  2115#  2128#  2129#  2178#  2182#
                  2186#  2195#  2201#  2228#  2232#  2245#  2253#  2341#  2353#  2358#  2366#  2420#  2424#
                  2427#  2457#  2461#  2463#  2488#  2492#  2494#  2535#  2539#  2545#  2595#  2662#  2664#
                  2668#  2670#  2720#  2738#  2740#  3090#  3191#
TSTEST= 000043      17#  1126#  1148#  1170#  1192#  1212#  1248#  1292#  1321#  1347#  1371#  1398#  1441#
                  1477#  1511#  1552#  1589#  1630#  1671#  1718#  1755#  1810#  1835#  1871#  1908#  1936#
                  1974#  1978   1993   2008   2102#  2106   2119   2196#  2247#  2367#  2428#  2464#  2495#
                  2546#  2671#  3090   3422
TSTSTM= 177777      17#   465    467    469    471    473    509    513    522    573    586    587    588
                   603    604    612    616    624    628    900   1104   1105   1106   1113   1135   1136
                  1137   1138   1157   1158   1159   1160   1179   1180   1181   1182   1201   1202   1203
                  1204   1226   1227   1228   1232   1233   1249   1269   1273   1276   1281   1282   1294
                  1305   1306   1310   1311   1323   1332   1333   1337   1338   1349   1355   1356   1360
                  1361   1375   1382   1383   1387   1388   1400   1412   1413   1425   1426   1430   1431
                  1443   1453   1454   1462   1463   1467   1468   1479   1486   1487   1495   1496   1500
                  1501   1515   1523   1524   1533   1534   1538   1539   1563   1564   1569   1570   1577
                  1578   1604   1605   1610   1611   1618   1619   1645   1646   1651   1652   1659   1660
                  1687   1688   1693   1694   1699   1700   1707   1708   1734   1735   1739   1740   1743
                  1744   1777   1778   1785   1786   1787   1796   1797   1798   1812   1821   1822   1823
                  1839   1840   1854   1855   1859   1860   1861   1890   1891   1898   1911   1918   1919
                  1920   1921   1949   1956   1957   1978   1989   1993   2004   2008   2014   2015   2019
                  2025   2026   2029   2034   2035   2044   2045   2050   2051   2056   2057   2070   2072
                  2073   2081   2086   2087   2090   2106   2115   2119   2128   2129   2133   2140   2146
                  2147   2152   2153   2166   2168   2169   2177   2178   2179   2182   2201   2224   2228
                  2253   2265   2305   2310   2338   2340   2341   2353   2376   2383   2391   2399   2415
                  2420   2447   2452   2456   2457   2483   2487   2488   2518   2528   2530   2534   2535
                  2558   2595   2601   2620   2634   2644   2655   2662   2664   2683   2720   2724   2737
                  2738   2740   2769   2771   2773   2845   3012   3049   3051   3101   3104   3107   3110
                  3113   3116   3119   3122   3126   3133   3142   3149   3150   3157   3158   3161   3173
                  3174   3175   3176   3181   3188   3189   3190   3191   3404
TSTSTS= 000001      17#  1126#  1148#  1170#  1192#  1212#  1248#  1292#  1321#  1347#  1371#  1398#  1441#
                  1477#  1511#  1552#  1589#  1630#  1671#  1718#  1755#  1810#  1835#  1871#  1908#  1936#
                  1974#  2102#  2196#  2247#  2367#  2428#  2464#  2495#  2546#  2671#
TSSAU = 010011     620#   628
TSSAUT= 010006     577#   588
TSSCLE= 010007     592#   604
TSSDAT= 010066    3430#  3436
TSSDU = 010010     608#   616
TSSHAR= 010000      55#    66
TSSHW = 010001      81#    91
TSSINI= 010005     464#   573
TSSMSG= 010063    3096#  3191
TSSPC = 000001    3429#  3437
TSSPRO= 010004     456#
TSSPTA= 010065    3429#  3430#
```

```
T$$RPT= 010012          1099#  1107   1113
T$$SEG= 010000          1294#  1306   1310#  1323#  1333   1337#  1349#  1356   1360#  1375#  1383   1387#  1400#
                        1413   1426   1430#  1443#  1454   1463   1467#  1479#  1487   1496   1500#  1515#  1524
                        1534   1538#  1839#  1859#  2019#  2029   2087#  2133#  2178#  2265#  2341#  2601#  2662#
                        2724#  2738#
T$$SOF= 010002            97#   112
T$$SUB= 010052          1978#  1989   1993#  2004   2008#  2014   2106#  2115   2119#  2128
T$$SW = 010003           135#   143
T$$TES= 010062          1126#  1138   1148#  1160   1170#  1182   1192#  1204   1212#  1233   1248#  1282   1292#
                        1311   1321#  1338   1347#  1361   1371#  1388   1398#  1431   1441#  1468   1477#  1501
                        1511#  1539   1552#  1578   1589#  1619   1630#  1660   1671#  1708   1718#  1744   1755#
                        1787   1798   1810#  1823   1835#  1861   1871#  1898   1908#  1921   1936#  1957   1974#
                        2015   2090   2102#  2129   2182   2196#  2201   2228   2247#  2253   2353   2367#  2420
                        2428#  2457   2464#  2488   2495#  2535   2546#  2595   2664   2671#  2720   2740
T1      012114 G        1126#  3090
T10     013332 G        1371#  3090
T11     013436 G        1398#  3090
T12     013664 G        1441#  3090
T13     014050 G        1477#  3090
T14     014216 G        1511#  3090
T15     014410 G        1552#  3090
T16     014612 G        1589#  3090
T17     015036 G        1630#  3090
T18     015262 G        1671#  3090
T19     015552 G        1718#  3090
T2      012200 G        1148#  3090
T20     015746 G        1755#  3090
T21     016160 G        1810#  3090
T22     016246 G        1835#  3090
T23     016414 G        1871#  3090
T24     016550 G        1908#  3090
T25     016634 G        1936#  3090
T26     016762 G        1974#  3090
T26.1   016762          1978#
T26.2   017030          1993#
T26.3   017076          2008#
T27     017662 G        2102#  3090
T27.1   017662          2106#
T27.2   017714          2119#
T28     020306 G        2196#  3090
T29     020502 G        2247#  3090
T3      012264 G        1170#  3090
T30     021276 G        2367#  3090
T31     021650 G        2428#  3090
T32     022054 G        2464#  3090
T33     022222 G        2495#  3090
T34     022502 G        2546#  3090
T35     023364 G        2671#  3090
T4      012350 G        1192#  3090
T5      012434 G        1212#  3090
T6      012556 G        1248#  3090
T7      013020 G        1292#  3090
T8      013140 G        1321#  3090
T9      013244 G        1347#  3090
UAM   = 000200 G         147#
UNITST  003322           223#   517*   518*   520    522    536    578    587    1136   1158   1180   1202   1227
```

```
                    3189
VECA    002340        62     74#
WDELAY  010160       596    743#    895   1260   1263   2500
WRCHK = 000002       174#  2725
WRITCH  023650      2703   2708   2724#
WRITE = 000012       178#  2502
WTCRDY  010640       892#  1765   1817   1846   1884   1915   1944   2022   2143   2215   2285   2443   2479
                    2517   2520   2614   2640   2731
XFIFO   017754      2114   2127   2133#
XPOLY   003404       248#   965    976
X$ALWA= 000000        17#    59     64
X$FALS= 000040        17#    57    101
X$OFFS= 000400        17#    57     59     64    101    104
X$TRUE= 000020        17#   104
.       = 040362      35#    44#    57     59     64     77#   101    104    131#   274#   448#   449#   615#
                    627#   666    674    682    803   1107  1112#  1306   1333   1356   1383   1413   1426
                   1454   1463   1487   1496   1524   1534   1787   2015   2029   2129   2201   2253   2595
                   2720   2895   3069#  3300#  3350   3411#  3412#  3417   3418#  3430   3437
.ABUSW  011240     1001#  1015*  1064*  1067*  1106   2209   2255
.LSTAD  011236      999#  1085*
.LSTPG  011234      997#  1084*  1101
.MSIZE  011232      996#  1086*  1105   2199
.SIZE   011242      508   1015#
.SIZKT  011370     1028   1042#
.SIZME  011532     1070#
.SIZXI  011604     1040   1084#
```

BCOMPL 466 468 470 472 474 3182
BEGIN. 211# 1126 1148 1170 1192 1212 1248 1292 1321 1347 1371 1398 1441 1477 1511
1552 1589 1630 1671 1718 1755 1810 1835 1871 1908 1936 1974 2102 2196 2247
2367 2428 2464 2495 2546 2671
BGNAU 620
BGNAUT 577
BGNCLN 592
BGNDU 608
BGNHRD 55
BGNHW 81
BGNINI 464
BGNMSG 3096
BGNPRO 456
BGNPTA 3430
BGNRPT 1099
BGNSEG 1294 1323 1349 1375 1400 1443 1479 1515 1839 2019 2133 2265 2601 2724
BGNSET 3429
BGNSFT 97
BGNSUB 1978 1993 2008 2106 2119
BGNSW 135
BGNTST 1126 1148 1170 1192 1212 1248 1292 1321 1347 1371 1398 1441 1477 1511 1552
1589 1630 1671 1718 1755 1810 1835 1871 1908 1936 1974 2102 2196 2247 2367
2428 2464 2495 2546 2671
BNCOMP 523
CKLOOP 1564 1570 1605 1611 1646 1652 1688 1694 1700 1735 1740 1778 1786 1797 1855
1891 1919 2026 2035 2045 2051 2057 2073 2147 2153 2169 2340
DESCRI 44
DEVTYP 46
DFERR 204# 900 1232 1269 1273 1276 1281 1305 1332 1355 1382 1412 1425 1453 1462
1486 1495 1523 1533 1563 1569 1577 1604 1610 1618 1645 1651 1659 1687 1693
1699 1707 1734 1739 1743 1777 1785 1796 1821 1854 1890 1918 1949 1956 2025
2034 2044 2050 2056 2070 2086 2146 2152 2166 2224 2305 2310 2338 2376 2383
2391 2399 2415 2447 2452 2456 2483 2487 2518 2528 2530 2534 2558 2620 2634
2644 2655 2683 2737 2769 2771 2773
DISPAT 3090
DISPLA 99
DOCLN 513 1137 1159 1181 1203 1228 3049 3190
DODU 587 1136 1158 1180 1202 1227 3189
DORPT 509
ENDAU 628
ENDAUT 588
ENDCLN 604
ENDDU 616
ENDHRD 66
ENDHW 91
ENDINI 573
ENDMSG 3191
ENDPRO 460
ENDPTA 3436
ENDRPT 1113
ENDSEG 1310 1337 1360 1387 1430 1467 1500 1538 1859 2087 2178 2341 2662 2738
ENDSET 3437
ENDSFT 112
ENDSUB 1989 2004 2014 2115 2128
ENDSW 143
ENDTST 1138 1160 1182 1204 1233 1282 1311 1338 1361 1388 1431 1468 1501 1539 1578

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

```
                1552#   1578#   1589#   1619#   1630#   1660#   1671#   1708#   1718#   1744#   1755#   1798#   1810#   1823#   1835#
                1859#   1861#   1871#   1898#   1908#   1921#   1936#   1957#   1974#   1978#   1989#   1993#   2004#   2008#   2014#
                2087#   2090#   2102#   2106#   2115#   2119#   2128#   2178#   2182#   2196#   2228#   2247#   2341#   2353#   2367#
                2420#   2428#   2457#   2464#   2488#   2495#   2535#   2546#   2662#   2664#   2671#   2738#   2740#   3090#   3096#
                3191#   3422#   3430#   3436#
MSGETS            57#     59#     64#     66#     91#    101#    104#    112#    143#    460#    573#    588#    604#    616#    628#
                1113#   1138#   1160#   1182#   1204#   1233#   1282#   1306#   1310#   1311#   1333#   1337#   1338#   1356#   1360#
                1361#   1383#   1387#   1388#   1413#   1426#   1430#   1431#   1454#   1463#   1467#   1468#   1487#   1496#   1500#
                1501#   1524#   1534#   1538#   1539#   1578#   1619#   1660#   1708#   1744#   1798#   1823#   1859#   1861#   1898#
                1921#   1957#   1989#   2004#   2014#   2029#   2087#   2090#   2115#   2128#   2178#   2182#   2228#   2341#   2353#
                2420#   2457#   2488#   2535#   2662#   2664#   2738#   2740#   3191#
MSGETT            57#     59#     64#    101#    104#   1107#   1306#   1333#   1356#   1383#   1413#   1426#   1454#   1463#   1487#
                1496#   1524#   1534#   1787#   2015#   2029#   2129#   2201#   2253#   2595#   2720#
MSGNGB            42#     44#     46#     55#     81#     97#    135#    456#    464#    577#    592#    608#    620#   1099#   3090#
                3096#   3422#
MSGNIN            42#     44#     46#     55#     56#     57#     58#     59#     60#     61#     62#     63#     64#     65#     66#
                  81#     97#     98#     99#    100#    101#    102#    103#    104#    105#    106#    107#    108#    109#    110#
                 111#    112#    135#    465#    466#    467#    468#    469#    470#    471#    472#    473#    474#    509#    513#
                 522#    523#    573#    586#    587#    588#    603#    604#    612#    616#    624#    628#    900#   1104#   1105#
                1106#   1107#   1113#   1135#   1136#   1137#   1138#   1157#   1158#   1159#   1160#   1179#   1180#   1181#   1182#
                1201#   1202#   1203#   1204#   1226#   1227#   1228#   1232#   1233#   1249#   1269#   1273#   1276#   1281#   1282#
                1294#   1305#   1306#   1310#   1311#   1323#   1332#   1335#   1337#   1338#   1349#   1355#   1356#   1360#   1361#
                1375#   1382#   1383#   1387#   1388#   1400#   1412#   1413#   1425#   1426#   1430#   1431#   1443#   1453#   1454#
                1462#   1463#   1467#   1468#   1479#   1486#   1487#   1495#   1496#   1500#   1501#   1515#   1523#   1524#   1533#
                1534#   1538#   1539#   1563#   1564#   1569#   1570#   1577#   1578#   1604#   1605#   1610#   1611#   1618#   1619#
                1645#   1646#   1651#   1652#   1659#   1660#   1687#   1688#   1693#   1694#   1699#   1700#   1707#   1708#   1734#
                1735#   1739#   1740#   1743#   1744#   1777#   1778#   1785#   1786#   1787#   1796#   1797#   1798#   1812#   1821#
                1822#   1823#   1839#   1840#   1854#   1855#   1859#   1860#   1861#   1890#   1891#   1898#   1911#   1918#   1919#
                1920#   1921#   1949#   1956#   1957#   1978#   1989#   1993#   2004#   2008#   2014#   2015#   2019#   2025#   2026#
                2029#   2034#   2035#   2044#   2045#   2050#   2051#   2056#   2057#   2070#   2072#   2073#   2081#   2086#   2087#
                2090#   2106#   2115#   2119#   2128#   2129#   2133#   2140#   2146#   2147#   2152#   2153#   2166#   2168#   2169#
                2177#   2178#   2179#   2182#   2201#   2224#   2228#   2253#   2265#   2305#   2310#   2338#   2340#   2341#   2353#
                2376#   2383#   2391#   2399#   2415#   2420#   2447#   2452#   2456#   2457#   2483#   2487#   2488#   2518#   2528#
                2530#   2534#   2535#   2558#   2595#   2601#   2620#   2634#   2644#   2655#   2662#   2664#   2683#   2720#   2724#
                2737#   2738#   2740#   2769#   2771#   2773#   2845#   3012#   3049#   3051#   3090#   3101#   3104#   3107#   3110#
                3113#   3116#   3119#   3122#   3126#   3133#   3142#   3149#   3150#   3157#   3158#   3161#   3173#   3174#   3175#
                3176#   3181#   3182#   3188#   3189#   3190#   3191#   3404#   3422#   3430#
MSGNLS          1310#   1337#   1360#   1387#   1430#   1467#   1500#   1538#   1859#   2087#   2178#   2341#   2662#   2738#
MSGNSU          1978#   1993#   2008#   2106#   2119#
MSGNTA            66#     91#    112#    143#    573#    588#    604#    616#    628#   1113#   1138#   1160#   1182#   1204#   1233#
                1282#   1311#   1338#   1361#   1388#   1431#   1468#   1501#   1539#   1578#   1619#   1660#   1708#   1744#   1798#
                1823#   1861#   1898#   1921#   1957#   1989#   2004#   2014#   2090#   2115#   2128#   2182#   2228#   2353#   2420#
                2457#   2488#   2535#   2664#   2740#   3191#   3430#   3436#
MSGNTE          1126#   1148#   1170#   1192#   1212#   1248#   1292#   1321#   1347#   1371#   1398#   1441#   1477#   1511#   1552#
                1589#   1630#   1671#   1718#   1755#   1810#   1835#   1871#   1908#   1936#   1974#   2102#   2196#   2247#   2367#
                2428#   2464#   2495#   2546#   2671#
MSHAPT            42#
MSHNAP            42#
MSINCR            55#     81#     97#    135#    456#    464#    465#    467#    469#    471#    473#    509#    513#    522#    573#
                 577#    586#    587#    588#    592#    603#    604#    608#    612#    616#    620#    624#    628#    900#   1099#
                1104#   1105#   1106#   1113#   1126#   1135#   1136#   1137#   1138#   1148#   1157#   1158#   1159#   1160#   1170#
                1179#   1180#   1181#   1182#   1192#   1201#   1202#   1203#   1204#   1212#   1226#   1227#   1228#   1232#   1233#
                1248#   1249#   1269#   1273#   1276#   1281#   1282#   1292#   1294#   1305#   1306#   1310#   1311#   1321#   1323#
                1332#   1335#   1337#   1338#   1347#   1349#   1355#   1356#   1360#   1361#   1371#   1375#   1382#   1383#   1387#
                1388#   1398#   1400#   1412#   1413#   1425#   1426#   1430#   1431#   1441#   1443#   1453#   1454#   1462#   1463#
                1467#   1468#   1477#   1479#   1486#   1487#   1495#   1496#   1500#   1501#   1511#   1515#   1523#   1524#   1533#
```

```
                1534#  1538#  1539#  1552#  1563#  1564#  1569#  1570#  1577#  1578#  1589#  1604#  1605#  1610#  1611#
                1618#  1619#  1630#  1645#  1646#  1721#  1652#  1659#  1660#  1671#  1687#  1688#  1693#  1694#  1699#
                1700#  1707#  1708#  1718#  1734#  1735#  1738#  1740#  1743#  1744#  1755#  1777#  1778#  1785#  1786#
                1787#  1796#  1797#  1798#  1810#  1812#  1821#  1822#  1823#  1835#  1839#  1840#  1854#  1855#  1859#
                1860#  1861#  1871#  1890#  1891#  1898#  1908#  1911#  1918#  1919#  1920#  1921#  1936#  1949#  1956#
                1957#  1974#  1978#  1989#  1993#  2004#  2008#  2014#  2015#  2019#  2025#  2026#  2029#  2034#  2035#
                2044#  2045#  2050#  2051#  2056#  2057#  2070#  2072#  2073#  2081#  2086#  2087#  2090#  2102#  2106#
                2115#  2119#  2128#  2129#  2133#  2140#  2146#  2147#  2152#  2153#  2166#  2168#  2169#  2177#  2178#
                2179#  2182#  2196#  2201#  2224#  2228#  2247#  2253#  2265#  2305#  2310#  2338#  2340#  2341#  2353#
                2367#  2376#  2383#  2391#  2399#  2415#  2420#  2428#  2447#  2452#  2456#  2457#  2464#  2483#  2487#
                2488#  2495#  2518#  2528#  2530#  2534#  2535#  2546#  2558#  2595#  2601#  2620#  2634#  2644#  2655#
                2662#  2664#  2671#  2683#  2720#  2724#  2737#  2738#  2740#  2769#  2771#  2773#  2845#  3012#  3049#
                3051#  3096#  3101#  3104#  3107#  3110#  3113#  3116#  3119#  3122#  3126#  3133#  3142#  3149#  3150#
                3157#  3158#  3161#  3173#  3174#  3175#  3176#  3181#  3188#  3189#  3190#  3191#  3404#  3429#  3430#
MSLDRO           465#   467#   469#   471#   473#   522#   587#   603#  1136#  1158#  1180#  1202#  1227#  1249#  1812#
                1822#  1840#  1860#  1911#  1920#  2140#  2179#  3189#
MSMCHI            17#
MSMCLO            17#
MSPOP             66#    91#   112#   143#   460#   573#   588#   604#   616#   628#  1113#  1138#  1160#  1182#  1204#
                1233#  1282#  1310#  1311#  1337#  1338#  1360#  1361#  1387#  1388#  1430#  1431#  1467#  1468#  1500#
                1501#  1538#  1539#  1578#  1619#  1660#  1708#  1744#  1798#  1823#  1859#  1861#  1898#  1921#  1957#
                1989#  2004#  2014#  2087#  2090#  2115#  2128#  2178#  2182#  2228#  2341#  2353#  2420#  2457#  2488#
                2535#  2662#  2664#  2738#  2740#  3191#
MSPRIN            612#   624#  1104#  1105#  1106#  2072#  2081#  2168#  2177#  2845#  3012#  3051#  3101#  3104#  3107#
                3110#  3113#  3116#  3119#  3122#  3126#  3133#  3142#  3149#  3150#  3157#  3158#  3161#  3173#  3174#
                3175#  3176#  3188#
MSPUSH             55#    81#    97#   135#   456#   464#   577#   592#   608#   620#  1099#  1126#  1148#  1170#  1192#
                1212#  1248#  1292#  1294#  1321#  1323#  1347#  1349#  1371#  1375#  1398#  1400#  1441#  1443#  1477#
                1479#  1511#  1515#  1552#  1589#  1630#  1671#  1718#  1755#  1810#  1835#  1839#  1871#  1908#  1936#
                1974#  1978#  1993#  2008#  2019#  2102#  2106#  2119#  2133#  2196#  2247#  2265#  2367#  2428#  2464#
                2495#  2546#  2601#  2671#  2724#  3096#
MSPUT             612#   624#  1104#  1105#  1106#  2072#  2081#  2168#  2177#  2845#  3012#  3051#  3101#  3104#  3107#
                3110#  3113#  3116#  3119#  3122#  3126#  3133#  3142#  3149#  3150#  3157#  3158#  3161#  3173#  3174#
                3175#  3176#  3188#
MSPUT1            612#   624#  1104#  1105#  1106#  2072#  2081#  2168#  2177#  2845#  3012#  3051#  3101#  3104#  3107#
                3110#  3113#  3116#  3119#  3122#  3126#  3133#  3142#  3149#  3150#  3157#  3158#  3161#  3173#  3174#
                3175#  3176#  3188#
MSRADI             56#    58#    60#    61#    62#    63#    65#    98#   100#   102#   103#   105#   106#   107#   108#
                 109#   110#   111#
MSRNRO             522#  3404#
MSSETS             55#    81#    97#   135#   456#   464#   577#   592#   608#   620#  1099#  1126#  1148#  1170#  1192#
                1212#  1248#  1292#  1294#  1321#  1323#  1347#  1349#  1371#  1375#  1398#  1400#  1441#  1443#  1477#
                1479#  1511#  1515#  1552#  1589#  1630#  1671#  1718#  1755#  1810#  1835#  1839#  1871#  1908#  1936#
                1974#  1978#  1993#  2008#  2019#  2102#  2106#  2119#  2133#  2196#  2247#  2265#  2367#  2428#  2464#
                2495#  2546#  2601#  2671#  2724#  3096#
MSSVC             465#   467#   469#   471#   473#   509#   513#   522#   573#   586   587#   588#   603#   604#   612#
                 616#   624#   628#   900  1104#  1105#  1106#  1107#  1113#  1135  1136#  1137#  1138#  1157  1158#
                1159#  1160#  1179  1180#  1181#  1182#  1201  1202#  1203#  1204#  1226  1227#  1228#  1232  1233#
                1249#  1269  1273  1276  1281  1282#  1294#  1305  1306#  1310#  1511#  1523#  1332  1333#  1337#
                1338#  1349#  1355  1356#  1360#  1361#  1375#  1382  1383#  1387#  1388#  1400#  1412  1413#  1425
                1426#  1430#  1431#  1443#  1453  1454#  1462  1463#  1467#  1468#  1479#  1486  1487#  1495  1496#
                1500#  1501#  1515#  1523  1524#  1533  1534#  1537  1538#  1539#  1563  1564#  1569  1570#  1577  1578#
                1604  1605#  1610  1611#  1618  1619#  1645  1646#  1651  1652#  1659  1660#  1687  1688#  1693
                1694#  1699  1700#  1707  1708#  1734  1735#  1739  1740#  1743  1744#  1777  1778#  1785  1786#
                1787#  1796  1797#  1798#  1812#  1821  1822#  1823#  1839#  1840#  1854  1855#  1859#  1860#  1861#
                1890  1891#  1898#  1911#  1918  1919#  1920#  1921#  1949  1956  1957#  1978#  1989#  1993#  2004#
```

MSTLAB
2008# 2014# 2015# 2019# 2025 2026# 2029# 2034 2035# 2044 2045# 2050 2051# 2056 2057#
2070 2072# 2073# 2081# 2086 2087# 2090# 2106# 2115# 2119# 2128# 2129# 2133# 2140# 2146
2147# 2152 2153# 2166 2168# 2169# 2177# 2178# 2179# 2182# 2201# 2224 2228# 2253# 2265#
2305 2310 2338 2340# 2341# 2353# 2376 2383# 2391 2399 2415 2420# 2447 2452 2456
2457# 2483 2487 2488# 2518 2528 2530 2534 2535# 2558 2595# 2601# 2620 2634 2644
2655 2662# 2664# 2683# 2720# 2724# 2737 2758# 2740# 2769 2771 2773 2845# 3012# 3049#
3051# 3101# 3104# 3107# 3110# 3113# 3116# 3119# 3122# 3126# 3133# 3142# 3149# 3150# 3157#
3158# 3161# 3173# 3174# 3175# 3176# 3181# 3188# 3189# 3190# 3191# 3404#
465# 467# 469# 471# 473# 509# 513# 522# 573# 586# 587# 588# 603# 604# 612#
616# 624# 628# 900# 1104# 1105# 1106# 1113# 1135# 1136# 1137# 1138# 1157# 1158# 1159#
1160# 1179# 1180# 1181# 1182# 1201# 1202# 1203# 1204# 1226# 1227# 1228# 1232# 1233# 1249#
1269# 1273# 1276# 1281# 1282# 1294# 1305# 1306# 1310# 1311# 1323# 1332# 1333# 1337# 1338#
1349# 1355# 1356# 1360# 1361# 1375# 1382# 1383# 1387# 1388# 1400# 1412# 1413# 1425# 1426#
1430# 1431# 1443# 1453# 1454# 1462# 1463# 1467# 1468# 1479# 1486# 1487# 1495# 1496# 1500#
1501# 1515# 1523# 1524# 1533# 1534# 1538# 1539# 1563# 1564# 1569# 1570# 1577# 1578# 1604#
1605# 1610# 1611# 1618# 1619# 1645# 1646# 1651# 1652# 1659# 1660# 1687# 1688# 1693# 1694#
1699# 1700# 1707# 1708# 1734# 1735# 1739# 1740# 1743# 1744# 1777# 1778# 1785# 1786# 1787#
1796# 1797# 1798# 1812# 1821# 1822# 1823# 1839# 1840# 1854# 1855# 1859# 1860# 1861# 1890#
1891# 1898# 1911# 1918# 1919# 1920# 1921# 1949# 1956# 1957# 1978# 1989# 1993# 2004# 2008#
2014# 2015# 2019# 2025# 2026# 2029# 2034# 2035# 2044# 2045# 2050# 2051# 2056# 2057# 2070#
2072# 2073# 2081# 2086# 2087# 2090# 2106# 2115# 2119# 2128# 2129# 2133# 2140# 2146# 2147#
2152# 2153# 2166# 2168# 2169# 2177# 2178# 2179# 2182# 2201# 2224# 2228# 2253# 2265# 2305#
2310# 2338# 2340# 2341# 2353# 2376# 2383# 2391# 2399# 2415# 2420# 2447# 2452# 2456# 2457#
2483# 2487# 2488# 2518# 2528# 2530# 2534# 2535# 2558# 2595# 2601# 2620# 2634# 2644# 2655#
2662# 2664# 2683# 2720# 2724# 2737# 2758# 2740# 2769# 2771# 2773# 2845# 3012# 3049# 3051#
3101# 3104# 3107# 3110# 3113# 3116# 3119# 3122# 3126# 3133# 3142# 3149# 3150# 3157# 3158#
3161# 3173# 3174# 3175# 3176# 3181# 3188# 3189# 3190# 3191# 3404#

MSTSTL
465# 467# 469# 471# 473# 509# 513# 522# 573# 586# 587# 588# 603# 604# 612#
616# 624# 628# 900# 1104# 1105# 1106# 1113# 1135# 1136# 1137# 1138# 1157# 1158# 1159#
1160# 1179# 1180# 1181# 1182# 1201# 1202# 1203# 1204# 1226# 1227# 1228# 1232# 1233# 1249#
1269# 1273# 1276# 1281# 1282# 1294# 1305# 1306# 1310# 1311# 1323# 1332# 1333# 1337# 1338#
1349# 1355# 1356# 1360# 1361# 1375# 1382# 1383# 1387# 1388# 1400# 1412# 1413# 1425# 1426#
1430# 1431# 1443# 1453# 1454# 1462# 1463# 1467# 1468# 1479# 1486# 1487# 1495# 1496# 1500#
1501# 1515# 1523# 1524# 1533# 1534# 1538# 1539# 1563# 1564# 1569# 1570# 1577# 1578# 1604#
1605# 1610# 1611# 1618# 1619# 1645# 1646# 1651# 1652# 1659# 1660# 1687# 1688# 1693# 1694#
1699# 1700# 1707# 1708# 1734# 1735# 1739# 1740# 1743# 1744# 1777# 1778# 1785# 1786# 1787#
1796# 1797# 1798# 1812# 1821# 1822# 1823# 1839# 1840# 1854# 1855# 1859# 1860# 1861# 1890#
1891# 1898# 1911# 1918# 1919# 1920# 1921# 1949# 1956# 1957# 1978# 1989# 1993# 2004# 2008#
2014# 2015# 2019# 2025# 2026# 2029# 2034# 2035# 2044# 2045# 2050# 2051# 2056# 2057# 2070#
2072# 2073# 2081# 2086# 2087# 2090# 2106# 2115# 2119# 2128# 2129# 2133# 2140# 2146# 2147#
2152# 2153# 2166# 2168# 2169# 2177# 2178# 2179# 2182# 2201# 2224# 2228# 2253# 2265# 2305#
2310# 2338# 2340# 2341# 2353# 2376# 2383# 2391# 2399# 2415# 2420# 2447# 2452# 2456# 2457#
2483# 2487# 2488# 2518# 2528# 2530# 2534# 2535# 2558# 2595# 2601# 2620# 2634# 2644# 2655#
2662# 2664# 2683# 2720# 2724# 2737# 2758# 2740# 2769# 2771# 2773# 2845# 3012# 3049# 3051#
3101# 3104# 3107# 3110# 3113# 3116# 3119# 3122# 3126# 3133# 3142# 3149# 3150# 3157# 3158#
3161# 3173# 3174# 3175# 3176# 3181# 3188# 3189# 3190# 3191# 3404#

MSWORD
42# 56# 57# 58# 59# 60# 61# 62# 63# 64# 65# 98# 99# 100# 101#
102# 103# 104# 105# 106# 107# 108# 109# 110# 111# 586# 900# 1107# 1135# 1157#
1179# 1201# 1226# 1232# 1269# 1273# 1276# 1281# 1305# 1332# 1355# 1382# 1412# 1425# 1453#
1462# 1486# 1495# 1523# 1533# 1563# 1569# 1577# 1604# 1610# 1618# 1645# 1651# 1659# 1687#
1693# 1699# 1707# 1734# 1739# 1743# 1777# 1785# 1787# 1796# 1821# 1854# 1890# 1918# 1949#
1956# 2015# 2025# 2029# 2034# 2044# 2050# 2056# 2070# 2086# 2129# 2146# 2152# 2166# 2201#
2224# 2253# 2305# 2310# 2338# 2376# 2383# 2391# 2399# 2415# 2447# 2452# 2456# 2483# 2487#
2518# 2528# 2530# 2534# 2558# 2595# 2620# 2634# 2644# 2655# 2683# 2720# 2737# 2769# 2771#
2773# 3090# 3430#

MSXFER
57# 59# 64# 101# 104#

```
POINTE    41
PRINTB   2081    2177    3101    3104    3107    3110    3113    3116    3119    3122    3126    3133    3142    3173    3174
         3175    3176
PRINTF    612     624    1104    1105    1106    2845    3012    3051    3188
PRINTX   2072    2168    3149    3150    3157    3158    3161
READEF    465     467     469     471     473
SETPRI    603    1249    1812    1822    1840    1860    1911    1920    2140    2179
SFERR     199#    586    1135    1157    1179    1201    1226
STARS    1120    1125    1142    1147    1164    1169    1186    1191    1208    1211    1237    1247    1286    1291    1315
         1350    1362    1346    1365    1370    1392    1397    1435    1440    1472    1476    1505    1510    1543    1551
         1582    1588    1623    1629    1664    1670    1712    1717    1748    1754    1802    1809    1827    1834    1865
         1870    1903    1907    1925    1935    1961    1973    2094    2101    2186    2195    2232    2245    2358    2366
         2424    2427    2461    2463    2492    2494    2539    2545    2668    2670
SVC        16#     17
XFER       59      64    1107#   1787#   2015#   2029#   2129#   2201#   2253#   2595#   2720#
XFERF      57     101
XFERT     104

. ABS.  040362     000     CON   RO    REL   GBL   D

 ERRORS DETECTED:  0

 CVRLBA.BIC,CVRLBA.LST/LI:TOC/CRF=SVC33/ML,CVRLBA.P11
 RUN-TIME:  36 37  4 SECONDS
 RUN-TIME RATIO: 327/78=4.1
 CORE USED:  28K  (55 PAGES)
```